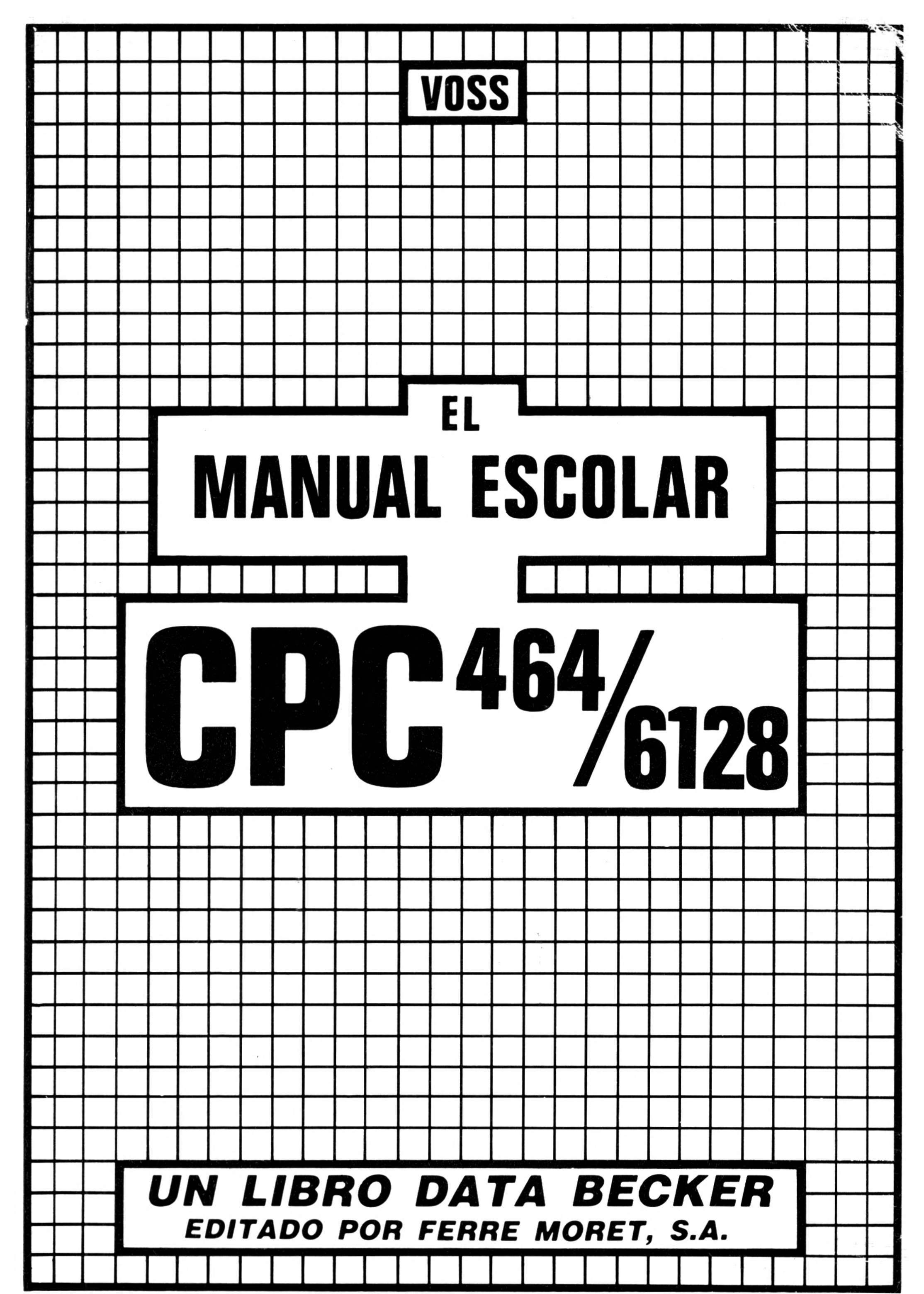

VOSS
EL
MANUAL ESCOLAR
CPC 464/6128
UN LIBRO DATA BECKER
EDITADO POR FERRE MORET, S.A.


VOSS
EL
MANUAL ESCOLAR
CPC 464/6128
UN LIBRO DATA BECKER
EDITADO POR FERRE MORET, S.A.

Este libro ha sido traducido por Dn. Joachim Hommen ingeniero electrónico, analista de sistemas y experto conocedor del CPC 464/6128.

Imprime : APSSA, ROCA UMBERT, 26 - L'HOSPITALET DE LL. (Barcelona)

ISBN 84-86437-15-6

Depósito Legal B-35.772/85

ESTRUCTURACION Página

==============

Página

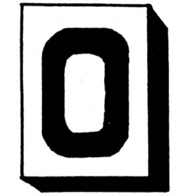
Página

Página

Advertencia importante

Los circuitos, procedimientos y programas reproducidos en este libro, son divulgados sin tener en cuenta el estado de las patentes. Están destinados exclusivamente al uso amateur o docente, y no pueden ser utilizados para fines comerciales.

Todos los circuitos, datos técnicos y programas de este libro, han sido elaborados o recopilados con el mayor cuidado por el autor y reproducidos utilizando medidas de control eficaces. No obstante, es posible que exista algún error. FERRE MORET,S.A. se ve por tanto obligada a advertirles, que no puede asumir ninguna garantía, ni responsabilidad jurídica, ni cualquier otra responsabilidad sobre las consecuencias atribuibles a datos erróneos. El autor les agradecerá en todo momento la comunicación de posibles fallos.



Prefacio

Los ordenadores como el AMSTRAND CPC 464/6128 no sólo están para jugar.Por muy fascinantes que puedan ser los programas elaborados - juegos de fútbol,guerras espaciales, carreras de coches, partidas de ajederez, luchas contra extraños monstruos, etc.-no hay nada más excitante que ser uno mismo el que programa.Quien analiza por primera vez un problema y consigue que el programa introducido lo resuelva en cuestión de segundos y sin errores, ése puede congratularse. Un logro de este tipo prueba la utilidad del ordenador, es decir, que nos libera del pesado trabajo mental, siendo ésta su principal aplicación, que determina su creciente importancia.

Por este motivo, en el presente libro queremos enseñarle al alumno cómo aplicar el ordenador, en concreto el AMSTRAND CPC 464/6128,para resolver los problemas fundamentales de la enseñanza escolar.
En resumidas cuentas, con este libro se persiguen simultáneamente cuatro objetivos:

1. Se toman los problemas escolares más normales, por ejemplo de matemáticas ,física u otras asignaturas y se proponen medios para una solución.

2. La descripción de estos medios es presentada al lector de forma que pueda aplicarlos al ordenador.

3. Se informará al lector sobre la utilización del lenguaje de programación BASIC destinado a aquellos programas necesarios para solucionar los problemas antes mencionados.

4. De esta forma, el lector dispondrá de una colección de programas, que le será útil en la escuela.

Comprenderá el lector que la utilidad de este libro se hace patente especialmente cuando prueba él mismo los programas presentados al igual que cuando ensaya con modificaciones de éstos o con otros programas.

Como en muchos otros ámbitos, dentro del campo de la programación y de la utilización de ordenadores se cumple lo siguiente:

'' La práctica hace al maestro ''

En los capítulos de este libro, se cubren los siguientes campos:

1. Problemas de matemáticas
2. Problemas de química
3. Problemas de física
4. Ejercicios de lenguas
5. Problemas de biología/ecología
6. Problemas de geografía e historia
7. Problemas de economía
8. Otros problemas

En estos capítulos se contienen algunas explicaciones acerca de las bases y condiciones de aplicación del ordenador y se discuten, también, los elementos básicos del lenguaje de programación BASIC.
Los lectores que posean suficientes conocimientos de BASIC pueden prescindir de estas explicaciones.
Debemos hacer la siguiente aclaración con respecto a los programas que desarrollamos y presentamos en los capítulos siguientes:

Por una parte, se trata de programas que deben resolver determinados problemas (Ejemplo: Programa para calcular un ''máximo común divisor'').

Estos programas pueden " alimentarse " con diferentes datos de partida, y sirven siempre para el mismo fin. Por otro lado, existen también programas sólo de entrenamiento: Por ejemplo, un programa que sirve para aprender vocablos ingleses, que no resuelve ningún problema, sino que sólo "entrena".
En todo caso, estos programas tienen aplicaciones más generales cuando se proveen de material de partida más variado.

Se están preparando una cassette y un diskette de programas para este libro, donde se encuentran todos los programas aquí discutidos, de modo que el lector que quiera utilizarlos se pueda ahorrar el trabajo de teclearlos. No obstante, no se incluyen los pequeños programas de muestra del capítulo siguiente, que sólo sirven para ilustrar los elementos básicos del lenguaje de programación BASIC.
La siguiente observación es de gran importancia para las próximas explicaciones:

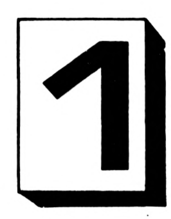
En los programas donde se resuelven los problemas del colegio hemos prescindido conscientemente de utilizar los elementos más avanzados y complejos del lenguaje BASIC. Por el contrario, nos hemos esforzado en utilizar los elementos básicos y sencillos del BASIC. El motivo es evitar sobrepasar, tanto las posibilidades del lector como las del principiante, en materia de programación y conseguir, de este modo, que tenga la ocasión de centrarse en los problemas.

Esta limitación tiene como consecuencia el que algunos programas parezcan demasiado simples pero, aun con los medios más sencillos funcionan y éste era nuestro objetivo.

No obstante, los programas más complejos de los últimos capítulos, requieren la aplicación de otros elementos sintácticos, que serán tratados en caso concreto.

Rogamos también al lector tenga en cuenta que algunos de los programas presentados figuran simplemente a modo de ejmplos:

Si, por ejemplo, en un programa de repaso de vocablos sólo se prevén 10, el lector, para poder utilizarlo de forma concreta, deberá ampliar primero la lista de éstos. Este procedimiento naturalmente no modifica la estructura básica del programa - que es lo que a fin de cuentas importa.



Capítulo 1 : Elementos básicos del lenguaje BASIC

1.1 Consideración previa

En este capítulo, se presentan y se explican además de algunos conceptos básicos del proceso de datos, que volverán a aparecer más tarde, los elementos básicos más importantes del lenguaje BASIC.
Aquellos lectores que ya conozcan suficientemente el BASIC pueden saltarse este capítulo.

Sin embargo, los lectores que no tengan ninguna experiencia como programadores pensarán que las explicaciones de este capítulo son algo someras. A éstos les diremos que en los próximos capítulos se aclarará la utilización, el funcionamiento y el efecto de estos elementos sintácticos mientras se aplican de forma práctica.
Visto de este modo, este capítulo contiene sólo una introducción.

1.2 Conceptos básicos que deberían conocerse

Un ordenador moderno no sólo sabe calcular. Sabe, asimismo, manipular letras, palabras y textos y puede, entre otras cosas, generar representaciones gráficas, producir música así como ejercer tareas de control. Ello significa que los ordenadores, que hoy en día ya pueden adquirirse por un precio bastante módico, no sólo disponen de la misma capacidad de cálculo que un ordenador grande de los años sesenta, que era económicamente inaccesible, sino que también su aplicación es más universal.

Datos : Bajo "datos" puede entenderse en el sentido más amplio la palabra "informaciones". Concretamente puede tratarse de:
- cifras y números
- valores
- letras
- símbolos (caracteres especiales)
- palabras y textos

Proceso : Bajo "proceso de datos" se entienden los procedimientos (frecuentemente cálculos), que sirven para registrar estos datos, para almacenarlos, para valorarlos o analizarlos y para ofrecer resultados.

Cuando el proceso de datos en la forma descrita no se realiza manualmente, sino con la ayuda de ordenadores, es necesario conocer otros conceptos nuevos:

bit : Bajo bit "binary information digit" se entiende la unidad de información más pequeña (ver antes : "datos" = "informaciones"). La unidad de información más pequeña se llama bit, porque un bit puede tener únicamente dos contenidos distintos. Visto desde la perspectiva del fabricante de ordenadores esto significa: un bit puede almacenar sólo dos contenidos de información. Estos contenidos se denominan usualmente 0 y 1.

byte : El byte es la reunión de una serie de bits. Normalmente son ocho los bits que constituyen un byte. De este modo, un byte está en condiciones de almacenar una serie de 8 ceros y/o unos.

Carácter : Distinguimos tres grupos de caracteres

1. caracteres numéricos = cifras
2. caracteres alfabéticos = letras
3. otros caracteres = caracteres especiales

Generalmente, cada uno de estos caracteres se representa en una serie de 8 ceros y/o unos en el ordenador, es decir, que un byte acepta un carácter.

Campo : Una serie de caracteres (p.e. palabras o números) constituyen un campo.

La palabra PRINT, por ejemplo, consta de 5 caracteres, ocupa 5 bytes en el ordenador, y como estos 5 símbolos pertenecen a una misma palabra, constituyen un campo.

Lo mismo ocurre con el número 178, que es almacenado en un campo de tres posiciones (se deduce que también pueden existir campos de una sóla posición).

Variable: Una variable es una magnitud que puede tomar diversos valores - precisando - un nombre, bajo el cual pueden seleccionarse diversos valores (ver abajo).
Por ejemplo, bajo el nombre ''tamaño'' podemos hallar los valores 178, 185, 167 etc., detrás del nombre ''número de niños'' los valores 1, 2, 0, 4 etc., bajo el nombre "localidad" por ejemplo los valores 'Palencia', 'Alcántara', 'Palafrugell'.

Valor : Como puede deducirse de la definición anterior, bajo "valor" entendemos una determinada característica que puede adquirir una variable dada. No es necesario que sean números, también puede tratarse de palabras o textos como en el ejemplo de la variable anterior "localidad".

Cada uno de estos valores ocupa un campo del ordenador(ver arriba). También puede llamarse posición de memoria.

String : Llamamos "cadena" a un texto que puede representar un valor de una variable (ver arriba); genéricamente se trata de una cadena de caracteres.

Tabla : La totalidad de los valores que puede tomar una variable (números o palabras) se llama "tabla".

Registro: Varios campos relacionados constituyen un registro. Por ejemplo, todos los datos referentes a una persona (nombre, sexo, edad, domicilio, etc.) forman un registro. En este ejemplo, el registro es una serie de números y/o palabras relacionados entre sí.
Una serie de palabras solas puede también constituir un registro, por ejemplo la instrucción:

```
PRINT   "PEPE", "ANA", "MAXIMO"
```

El registro se denomina también "record".

Fichero: Varios registros relacionados forman un fichero. Obtendremos un fichero por ejemplo si confeccionamos los registros de todos los empleados de una empresa. Podemos obtener igualmente un fichero si contemplamos todas las instrucciones de un programa en conjunto
El fichero se llama también "file".

1.3 Referente al funcionamiento de los ordenadores

Ya explicamos lo que se entiende bajo 'proceso de datos'. De ello se deduce qué actividades (funciones) debe realizar un equipo de proceso de datos:

1. Recepción de datos (entrada de datos)		
2. Almacenamiento de datos	E	entrada
3. Proceso de datos propiamente dicho (p.e. cálculos o traslado de textos y parecidos)	P	proceso
4. Salida de datos	S	salida

La entrada de información, su procesado y su salida hacen necesarios una serie de pasos organizativos , que no es necesario tratar aquí, porque generalmente los resuelve el mismo ordenador. De ello se encarga el llamado sistema operacional.
Así, el ordenador debe obtener nuestras informaciones, para que el proceso de los datos pueda ponerse en marcha y proporcionar resultados.

Las informaciones que debemos darle se clasifican en tres grupos:

1. Los datos a procesar
2. Las instrucciones
   (Pasos, que son necesarios para someter los datos introducidos a los procedimientos de valoración requeridos)
3. Comandos
   (Informaciones e instrucciones que damos al sistema operacional del ordenador; ver arriba)

De acuerdo con los diferentes grupos de cometidos que debe resolver el ordenador distinguiremos entre:

1. Aparatos de entrada (p.e. el teclado)
2. Ambito del procesado (la llamada unidad central)
3. Aparatos de salida (pantalla o impresora)

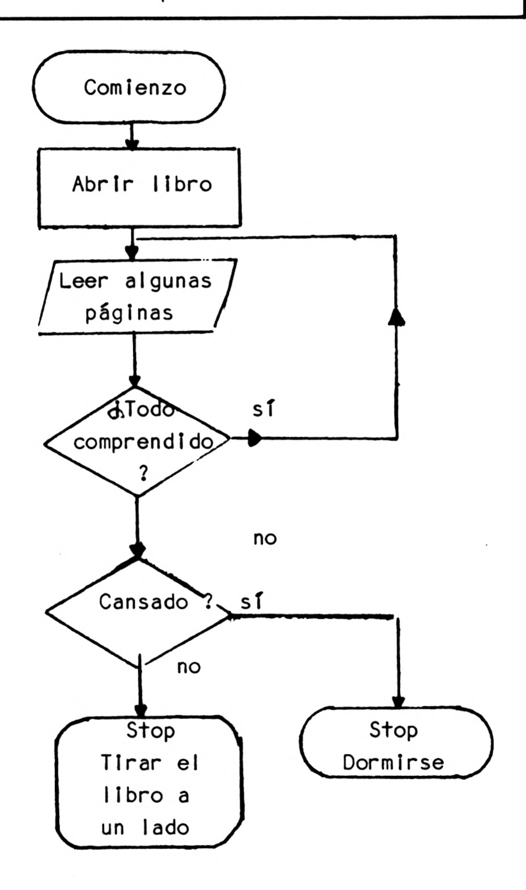
Aparte de esto existen también las llamadas memorias externas (o periféricas). Sirven para almacenar a largo plazo los programas y los datos. Generalmente se trata de la cassette y de los llamados diskettes, que se reproducen o leen en los correspondientes aparatos reproductores.

1.4 Análisis de problemas

Resolver un problema concreto con ayuda de un ordenador sólo es posible si se ha hallado de antemano su solución. El lector deberá recordar siempre esta evidencia. Esto quiere decir que la solución, a pesar de la modernidad de los ordenadores, sólo puede hallarse mentalmente y no en la máquina; el ordenador únicamente se ocupa de realizar los pasos necesarios.

Para que un ordenador pueda efectuar esta tarea debe indicársele el procedimiento de resolución en forma de instrucciones individuales. Esto lleva consigo que deba descomponerse mentalmente el problema a resolver en una sucesión lógica de pasos individuales - este procedimiento se denomina análisis del problema.

En un análisis de problema resulta útil representar la sucesión de pasos gráficamente en forma de un diagrama de flujo.
Veamos el siguiente ejemplo ilustrativo:



Fuente : W. VOSS : BASIC aprendido rápida- y fácilmente con el alphaTronic PC, Darmstadt 1984, (Heim-verlag), pag. 8.

En los capítulos siguientes utilizaremos una y otra vez el instrumento ''análisis del problema'', porque de este modo, la programación, es decir, la redacción de las secuencias de instrucciones para el ordenador, se facilita considerablemente, evitándose los errores de programación.

1.5 Visualización de resultados

Se le denomina programa a una sucesión de instrucciones que hacen que un ordenador resuelva por sí mismo un problema dado. Debe ser escrito en un lenguaje que el ordenador pueda comprender. Este es el BASIC.

BASIC =

Beginner's
All-Purpose
Symbolic
Instruction
Code

Todo programa BASIC consta de una sucesión de instrucciones (statements) que deben ser introducidos línea a línea en el ordenador. Una línea se llama sentencia.

Regla 1: Cada sentencia BASIC requiere un número.

Regla 2: Cada sentencia debe concluirse pulsando la tecla RETURN.

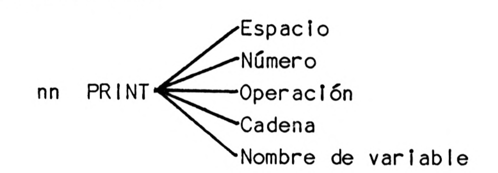
Regla 3: Una sentencia puede contener varias instrucciones, que hay que separar mediante dos puntos(:).

Instrucción 1:
----

nn END

La instrucción END cierra el programa BASIC.

Instrucción 2:



La instrucción PRINT trae los resultados a la pantalla. Se distinguen cinco posibilidades:

1. Espacio : Se obtiene una línea en blanco
2. Número : Se visualiza el número
3. Operación : Se visualiza el resultado de una operación
4. Cadena : Se visualiza la cadena (cadena de caracteres puesta entre comillas)
5. Nombre de variable : El contenido de este campo de la memoria es visualizado

La palabra PRINT puede abreviarse con un ?.

El siguiente programa BASIC utiliza todas estas posibilidades:

```
10 PRINT 3
20 PRINT
30 PRINT 4.8/2
40 ?
50 ? "SUSI"
60 ?
70 ?X
80 END
```

Este programa BASIC sólo se ejecuta si tecleamos el comando RUN:

Comando 1:

RUN

Para el ejemplo anterior, después del comando RUN obtendremos en pantalla los siguientes resultados:

```
3

2.4

SUSI

0
```

Regla 4: Los comandos no reciben ningún número de sentencia

Si después de la impresión del resultado deseamos ver nuevamente nuestro programa en pantalla, necesitamos el siguiente comando:

Comando 2:

LIST

La instrucción PRINT requiere algunas puntualizaciones adicionales:

Las diferentes posibilidades enunciadas en la instrucción 2 pueden utilizarse tanto mezcladas entre sí como en una sola instrucción PRINT.
En este caso deben separarse mediante comas y puntos .

Regla 5: Si se separan con comas las diferentes expresiones dentro de la instrucción PRINT, la impresión prosigue por el principio de la siguiente zona de la pantalla; cuando se utiliza el punto y coma, la impresión se produce justo a continuación.

Regla 6: Por regla general, toda instrucción PRINT produce un salto de una línea. Si la instrucción contiene la palabra PRINT sola, se produce un único salto de línea.

Regla 7: Una coma o un punto y coma al final de una instrucción PRINT reprimen el salto de línea.

La instrucción PRINT produce salidas de resultados que comienzan en el borde izquierdo de la pantalla. Para cambiar este hecho puede utilizarse la función TAB (tabulador).

```
PRINT TAB(10) 8
```

Esta instrucción produce la salida del resultado 8 a partir de la posición 10.

Para ello debe imaginarse la pantalla dividida en cuarenta columnas y 24 líneas, situándose la línea 0 arriba del todo y la columna 0 a la izquierda.

En el siguiente punto trataremos el concepto del nombre de variable, que ya jugaba un papel en la instrucción PRINT, además hablaremos sobre particularidades de las operaciones realizables.

Para borrar un programa que ya no nos es de utilidad, antes de introducir uno nuevo, se requiere el comando siguiente, que borra el contenido de la memoria del programa:

Comando 3:

```
NEW
```

1.6 Asignación de valores.

Como ya tendremos ocasión de ver, resulta muy útil para su aplicación, el dotar de nombres a las posiciones de memoria, con el fin de almacenar diversos valores bajo estos nombres.

En principio basta con distinguir entre 3 tipos de variable:

1. Variables reales
2. Variables de números enteros
3. Variables de string

Variables Reales pueden tomar números reales como valores, como por ejemplo 3.5 -17.01 18. 0.04 -7.5 etc. (en lugar de la coma decimal, al programar se utiliza un punto).

Variables de números enteros pueden tomar números enteros como valores, por ejemplo 4 8 -3 0 20 etc.

Variables de string tienen secuencias de caracteres como valor y deben colocarse entre comillas, por ejemplo "ANTONIO" "00" "SEVILLA" "B-30" etc.

Regla 8: Los nombres de las variables numéricas constan de 1 ó 2 caracteres alfanuméricos; el primero debe ser una letra. Los nombres de variables de números enteros llevan además un símbolo %, los nombres de las variables de cadena deben acompañarse de un signo $.

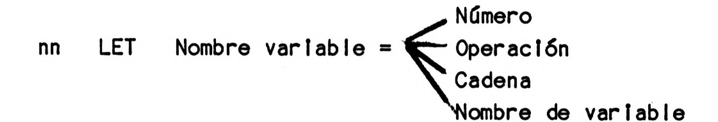
En muchos ordenadores los nombres pueden ser más largos (consultar el manual del usuario)

Ejemplos:
----

1. Variables reales	X	B1	NN
2. Variables de números enteros	C%	Y1%	AA%
3. Variable de cadena	F$	C2$	ZZ$

Después de aclarar estos conceptos hablaremos sobre la instrucción LET, que sirve para asignar valores de modo que pueda disponerse de los mismos a lo largo de todo el programa.

Instrucción 3:



Puede omitirse también la palabra LET.

Esta instrucción permite calcular de la misma manera que se hacía con la instrucción PRINT.
Los operadores para calcular son los siguientes:

+ Adición
- Sustracción
* Multiplicación
/ División
↑ Potenciación

El rango de los operadores es análogo al que conocemos de la escuela; en caso necesario pueden colocarse paréntesis.

Así, la instrucción LET nos permite escribir sentencias más complicadas. En caso de error a la hora de teclear, la forma más sencilla para corregir, es cerrar la sentencia errónea con RETURN y escribir simplemente la frase de nuevo. Para borrar del todo una sentencia, hay que teclear su número y pulsar RETURN.

Para insertar a posteriori una sentencia en un programa, es necesario introducir ''números de sentencia intermedios''.

Muchos cálculos pueden simplificarse mediante las llamadas funciones incorporadas. Las más importantes son:

Formato generalizado:
    Nom. Variable = Nombre de la función (argumento)

Funciones importantes

Nombre	Cometido
SIN COS TAN	Calcula los valores de la función angular, siendo el argumento en unidades de parámetro circular PI.
LOG	Logaritmo neperiano
EXP	Función inversa de la anterior
SQR	Raíz cuadrada
ABS	Valor absoluto
INT	Mayor parte entera existente en un número real
RND	Generación de números aleatorios
CHR$	Visualización de un carácter en código ASCII

Ahora podemos prescindir de los ejemplos. Le serán suficientes los capítulos siguientes.

1.7 Entrada de informaciones

Para introducir informaciones resulta muy adecuada la siguiente instrucción:

Instrucción 4:

```
nn   INPUT ("TEXTO";) lista de variables
```

Esta instrucción puede parecer complicada pero es fácil de comprender:

Cuando el ordenador llega a ésta instrucción, interrumpe el proceso del programa y produce un signo de interrogación en la pantalla (además del texto eventualmente indicado que puede ser asimismo omitido, por ello arriba figura entre paréntesis). Entonces espera a que el usuario introduzca tantos valores (separados por comas) como nombres existan en la lista de variables (es decir, como mínimo un valor, en el caso de que se trate de un sólo nombre).

Ejemplos:

INPUT X	El ordenador pide un número, que se almacenará en el campo X.
INPUT A,B,C	El ordenador pide 3 números.

INPUT "POR FAVOR ENTRAR NOMBRE"; N$: El ordenador visualiza el string entre comillas y pide una cadena (p.e. "ANTONIO"), que se almacenará en el campo N$.

Regla 9: Los nombres de la lista de variables de la instrucción INPUT se separan mediante comas.
Los valores a introducir también hay que separarlos de la misma forma.

Ejemplo:

```
10 INPUT "POR FAVOR CUATRO VALORES : ";A,B,C,D
20 LET S = A + B + C + D
30 LET AM = S/4
40 PRINT "PROMEDIO = "; AM
50 END
```

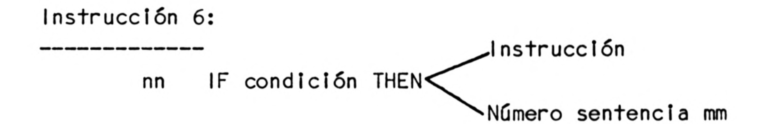
Vemos claramente como éste programa calcula el promedio de 4 valores a introducir.

1.8 Bifurcaciones del programa

Mientras en un programa BASIC no aparezca ninguna bifurcación (llamada salto), el proceso tiene lugar en el orden de los números de sentencia asignados. Si no se desea respetar este orden son necesarios los saltos en el programa.
Distinguimos entre saltos condicionales y saltos no condicionales.

Veámos en primer lugar el salto condicional, que se realiza gracias a la siguiente instrucción:

Instrucción 6:



Esta instrucción obliga al sistema operacional del ordenador a consultar: Si la condición expresada después del término IF se cumple, entonces se ejecuta la instrucción escrita detrás de THEN o se pasa a la sentencia número mm.

De no cumplirse esta condición, el control del programa pasa a la sentencia que sigue a la sentencia IF.

Regla 10: Si se cumple la sentencia IF, se efectúa la instrucción detrás del THEN o se salta al número de sentencia ahí escrito.
De no cumplirse esta condición, se procesa la siguiente sentencia.

Regla 11: Si hay otras instrucciones detrás del IF dentro de la misma sentencia, sólo se se realizarán si se cumple la condición.

Ejemplo:

```
10 I=1
20 = Q=I*I
30 W=SQR(I)
40 PRINT I,Q,W
50 I=I+1
60 IF I <=20 THEN 20
70 END
```

Este programa calcula los cuadrados (que aparecen en el campo Q; línea 20) de todos los números enteros del 1 al 20 (que aparecen en el campo I), y también sus raíces cuadradas (en el campo W;línea 30), imprimiendo cada grupo de tres valores en una línea de la pantalla.

La línea 60 resulta decisiva. Dicta que el programa vuelva una y otra vez a la línea 20, mientras en el campo I siga habiendo un valor menor de 21 tras elevar su contenido cada vez en 1 (línea 50).

En tanto esta condición se cumpla, se ejecutará la instrucción que sigue al THEN (que es el salto de vuelta a la línea 20); cuando ya no se cumpla se procesará la instrucción que sigue al IF (línea 70).

La alternativa al salto condicional la encarna el salto no condicional:

Instrucción 6:

```
nn  GOTO  mm
```

Cuando el control del programa alcanza ésta instrucción tiene lugar un salto a la sentencia número mm.

```
10 CLS
20 INPUT "INTRODUCIR VALOR:  "; X
30 N = N + 1
40 S = S + X
50 INPUT "OTRO VALOR (S/N) "; A$
60 IF A$ = "S" THEN 20
70 AM = S/N
80 PRINT "PROMEDIO = "; AM
90 END
```

Este programa calcula la media aritmética de una cantidad cualquiera de valores a introducir. Después de introducir un valor, el ordenador pregunta al usuario si desea introducir otros. Si el usuario contesta "S" ("SI") tiene lugar un salto de la línea 60 a la línea 20 y se pide el siguiente valor.
Si, por el contrario, ya no quedan valores, es decir, si el usuario ya no contesta con "S" en la instrucción INPUT de la línea 50, prosigue la ejecución del programa a partir de la línea 70 (cálculo y visualización de la media).

1.9 Bucles

A menudo resulta adecuado ejecutar determinadas porciones del programa más de una vez. Para ello se necesitan los llamados bucles (loops). Estos bucles pueden generarse sencillamente con las instrucciones tratadas hasta el momento, tal como puede apreciarse en el programa anterior.

Las siguientes instrucciones simplifican la constitución de bucles, abriéndose las siguientes posibilidades adicionales.

Instrucción 7: nn FOR nombre variable índice = valor inicial TO valor final (STEP longitud)

La siguiente instrucción va unida a la anterior

Instrucción 8: nn NEXT Nombre variable índice

La variable de índice que aparece en ambas instrucciones debe ser un número real. El valor inicial, el valor final y la longitud pueden ser números, variables o expresiones aritméticas.

''Cuando el ordenador llega a la instrucción FOR, se iguala la variable indexada al valor inicial y luego se ejecuta esta parte del programa situada entre FOR y NEXT con este valor.
Tras alcanzar el NEXT, la variable indexada salta al siguiente valor, que resulta del valor inicial + la longitud y se ejecuta nuevamente la misma parte del programa, esta vez con el segundo valor de la variable indexada.

Esto sigue hasta que, con una nueva adición de la longitud tras alcanzar el NEXT, se supera el valor final. Entonces se ejecuta la línea que sigue al NEXT.

Puede suprimirse el "STEP longitud" cuando la longitud deba ser igual a uno.''
(PRUST, 1982, pag. 64).

El programa siguiente imprime 10 veces la expresión "BUENOS DIAS" de forma uniforme, puesta una debajo de la otra:

```
10 FOR I=1 TO 10
20 PRINT "BUENOS DIAS"
30 NEXT I
40 PRINT
50 PRINT "FINAL"
60 END
```

1.10 La utilización de memorias externas

Siempre se desea guardar por un espacio de tiempo cualquier programa elaborado, de modo que se pueda utilizar en caso de necesidad (del mismo modo en ocasiones se desean utilizar programas de otros autores). Esto es posible utilizando cintas de cassette o diskettes. Para ello deben conocerse los siguientes comandos.

Para CARGAR un programa desde un cassette necesitamos el:

Comando 4:

```
LOAD "Nombre del programa"
```

Después de la indicación del ordenador, hay que manejar el casette, teniendo en cuenta que la cinta no debe estar adelantada de manera que no se encuentre el principio del programa.

Para grabar un programa en el cassette necesitamos el:

Comando 5:

SAVE "Nombre del programa"

Para poder encontrar en un cassette el programa que deseamos cargar, debemos entonces utilizar el comando:

Comando 6:

CAT

En la pantalla se visualizará el directorio, si el usuario ha seguido las instrucciones del ordenador.

1.11 Otros aspectos

Para cerrar este capítulo introductorio veamos unos pocos y útiles aspectos adicionales.

La presentación de un programa queda muy lograda si durante la ejecución del mismo y antes de la salida de los resultados, se borra la pantalla. Esto es posible gracias a la instrucción:

Sentencia 9:

```
nn CLS
```

Frecuentemente resulta necesario parar el proceso de un programa en casos como la programación del llamado bucle sinfín. Pulsando la tecla ESC se puede detener momentáneamente la ejecución que se reinicia pulsando otra tecla cualesquiera.
Si se pulsa 2 veces la tecla ESC, el programa queda interrumpido.

Finalmente queremos remarcar la posibilidad de interrumpir el proceso desde el propio programa con:

Sentencia 10:

```
nn STOP
```

Un programa interrumpido de esta forma puede continuar su ejecución a través del comando:

Comando 7:

```
CONT
```

Finalmente quisiera añadir que a menudo resulta útil incluir observaciones explicativas en un programa (títulos, comentarios, explicaciones, etc). Esto es posible con la siguiente instrucción:

Sentencia 11:

```
nn   REM   Texto
```

Esta sentencia no influye en la ejecución del programa, pero aparecerá en cualquier listado que se efectúe del mismo.

Espero que las presentes indicaciones sirvan para que, incluso el principiante, tenga una idea de los elementos básicos del lenguaje BASIC. De esta manera estará en condiciones de comprender las aplicaciones que siguen. Si aún así tuviera problemas de comprensión, sólo puedo darle un consejo:

P R O B A R

Probando los programas e intentando analizar por qué el ordenador hace precisamente una cosa y no otra se aprende más que de cualquier otra forma.

2

Capítulo 2. : Matemáticas

2.1 Consideración previa

Si queremos una buena aplicación para el ordenador, no hay nada más indicado que las matemáticas. Por esta razón empezaremos por ellas. Más tarde veremos, no obstante, que las matemáticas no son el único campo de aplicación.

En todo caso, buscaremos unos ejemplos adecuados pues todos sabemos que las matemáticas cubren un espectro de grados de dificultad muy amplio - empezando con las operaciones básicas y pasando por el cálculo infinitesimal—.

Tratarmos problemas de "dificultad media", es decir, nos orientaremos hacia problemas como los que se presentan en el último nivel de EGB y en el primer curso de bachillerato.

Las matemáticas son perfectas para demostrar las posibilidades del ordenador, sin que la presentación de los problemas tratados ocupe demasiado tiempo debido a un eventual grado de complejidad de los mismos.

Este libro no pretende exceder en las posibilidades del lector con los ejemplos de aplicación, simplemente se trata de presentar la capacidad del ordenador. Esto se conseguirá con problemas de estructura sencilla, adaptados a los problemas de los escolares más jóvenes.

En los programas de este y de los siguientes capítulos se seguirá la siguiente norma :

¡En los programas prescindiremos de corregir los errores de introducción del usuario!

Por ejemplo, en un problema matemático dado, si sólo pueden utilizarse números positivos (por ejemplo en las raíces), el correspondiente programa no tiene previsto el informarle al usuario en caso de haber introducido por error un número negativo.

Comenzaremos ahora con un ejemplo de grado de dificultad "medio".

2.2 Teorema de Pitágoras

Por medio del famoso teorema de Pitágoras puede mostrarse de forma clara, cómo hay que estructurar un problema.

Siempre se seguirán los siguientes pasos de trabajo:

Pasos de trabajo:

1. PASO : Presentación del problema

2. PASO : Análisis del problema

3. PASO : Diagrama de flujo

4. PASO : Programa

5. PASO : Lista de variables

6. PASO : Descripción del programa

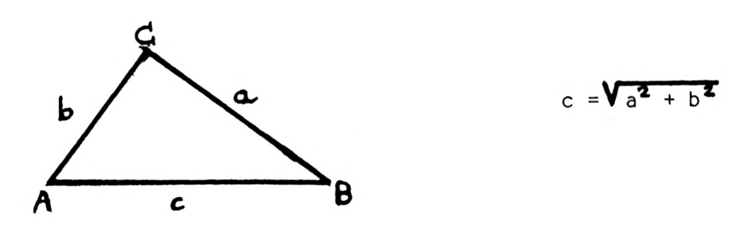
7. PASO : Resultados

Siguiendo esta estructuración pasemos ahora al problema citado:

1. PASO : Presentación del problema

El teorema de Pitágoras dice que en el triángulo rectángulo, la longitud del lado base (hipotenusa) es igual a la raiz cuadrada de la suma de los cuadrados de ambos catetos.

Esta figura ilustra lo dicho:



2. PASO : Análisis del problema

En este caso, el análisis del problema es muy sencillo:

Debe recordarse tanto aquí como en los siguientes problemas que cualquier proceso de datos funciona según el

Principio E P S

ya presentado.

E	=	Entrada
P	=	Proceso
S	=	Salida

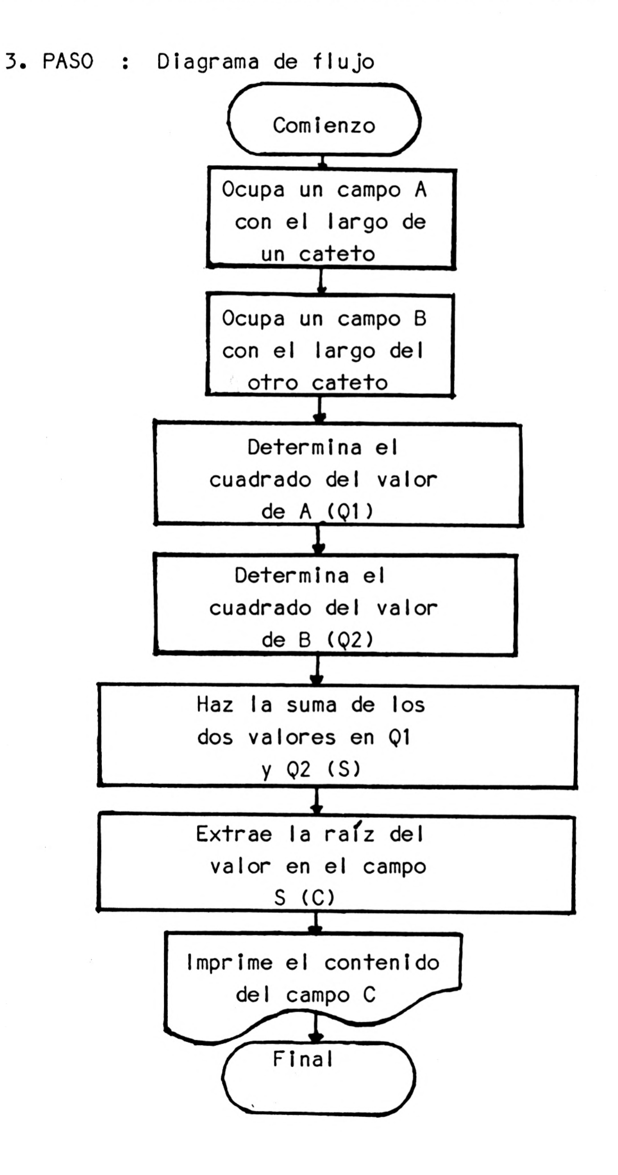
Por esta razón deberemos fijar, en primer lugar, las informaciones que debe recibir el ordenador de nuestra parte, para la resolución del problema.

Las informaciones de entrada son lógicamente las longitudes de ambos catetos a y b.

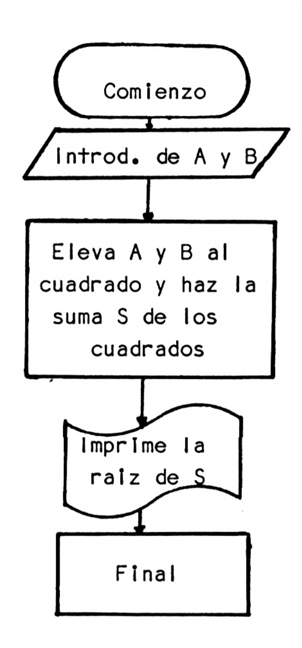
En el proceso, estos dos valores deben ser elevados al cuadrado; debe efectuarse su suma y finalmente extraerse la raíz cuadrada de esta suma.

La salida consiste únicamente en visualizar en la pantalla el resultado obtenido (la longitud de la hipotenusa c).

3. PASO : Diagrama de flujo



Versión resumida del diagrama de flujo:



4. PASO: Programa

```
10 REM PITAGORAS
20 CLS
30 PRINT "PROGRAMA PARA DETERMINAR LA HIPOTENUSA"
40 PRINT "DE UN TRIANGULO RECTANGULO."
50 PRINT : PRINT : PRINT TAB(10) "PROF. DR. W.VOSS 1984"
        : PRINT
60 INPUT "PRIMER CATETO   :"; A
70 INPUT "SEGUNDO CATETO  :"; B
80 Q1 = A * A
90 Q2 = B * B
100 S = Q1 + Q2
110 C = SQR (S)
120 PRINT : PRINT : PRINT "HIPOTENUSA C = "; C
130 END
```

Versión abreviada del programa

```
10 Hasta 50 : como arriba
60 INPUT "2 VALORES DE CATETOS : " ; A,B
70 C = SQR (A*A + B*B)
80 PRINT : PRINT "HIPOTENUSA C = ";C : END
```

5. PASO : LISTA DE VARIABLES

A	=	PRIMER CATETO
B	=	SEGUNDO CATETO
C	=	HIPOTENUSA
Q1	=	CUADRADO DEL PRIMER CATETO A
Q2	=	CUADRADO DEL SEGUNDO CATETO B
S	=	SUMA DE LOS CUADRADOS DE LOS CATETOS

6. PASO : Descripción del programa

Nos basamos en la primera versión, la más larga del programa:

Línea 10 : Comentario

Línea 20 : Se limpia la pantalla

Línea 30-50: Impresión del título en pantalla

Línea 60 : Se pide un valor para el primer cateto, que es almacenado en el campo A

Línea 70 : Lo mismo con el segundo cateto en el campo B

Línea 80 : Cálculo del cuadrado del primer cateto

Línea 90 : Cálculo del cuadrado del segundo cateto

Línea 100 : Suma de los dos cuadrados resultantes

Línea 110 : Cálculo de la raíz de la suma

Línea 120 : Impresión del resultado en pantalla

Línea 130 : Final del programa

7. PASO : Resultados

Una vez iniciado el programa (en su primera versión) aparece el título en la pantalla:

PROGRAMA PARA DETERMINAR LA HIPOTE-
NUSA DE UN TRIÁNGULO RECTANGULO

PROF. DR. W.VOSS, 1984

Dos líneas más abajo se pide el primer valor:

PRIMER CATETO : ?

A continuación, introduzcamos por ejemplo el valor 3 y el ordenador reaccionará de la forma siguiente:

SEGUNDO CATETO : ?

Introduzcamos por ejemplo el valor 4, a lo que el ordenador contestará:

HIPOTENUSA C = 5

Este programa es adecuado para calcular, a partir de cualquier valor, la hipotenusa de un triángulo rectángulo. En este sentido se trata de un programa muy general.

Hemos tratado este problema muy extensamente con el fin de ilustrar el procedimiento a seguir en cada uno de los ejemplos. De ahora en adelante nos ahorraremos explicaciones, por lo menos cuando se trate de ejemplos tan sencillos como el Teorema de Pitágoras.

2.3 M.C.D. y m.c.m.

1. PASO : Presentación del problema

En muchos cálculos algebráicos resulta útil conocer el máximo común divisor (M.C.D.) y el mínimo común múltiplo (m.c.m.) de diferentes números, especialmente cuando aparecen quebrados.

El M.C.D. es el valor que es divisor, sin resto, de dos números, sin que haya ningún otro número mayor que cumpla esta condición.

El m.c.m. es el número divisible entre los dos números de partida no habiendo otro número más pequeño para el que se cumpla esto.

Si tomamos por ejemplo los dos números Z1=30 y Z2=40, el M.C.D. es 10 y el m.c.m. es 120. (El lector lo puede comprobar mediante cálculo mental).

2. PASO : Análisis del problema

¿Cómo hallar el M.C.D. y el m.c.m. de dos números dados?

Analicemos por medio de los números 30 y 40 cuál es la forma de proceder más adecuada para hallar el M.C.D.:

1. Paso : 40 : 30 = 1 resto 10

2. Paso : 30 : 10 = 3 resto 0

En este ejemplo se halla el M.C.D. de la forma siguiente:

1. Paso : División del número mayor entre el número menor; anotación del resultado.

2. Paso : División del número menor entre el resto; anotación del nuevo resto.

Si el nuevo resto es 0, el divisor que nos ha llevado al mismo es el M.C.D. buscado

Veámos otro ejemplo ilustrativo:

Campo 1 : Z1 (Número mayor)	Campo 2 : Z2 (Número menor)	División	Resto
78	42	1	36
42	36	1	6
36	6	6	0

El M.C.D. de 78 y 42 es 6.

El esquema de cálculo puede transformarse fácilmente en un diagrama de flujo tal como muestra la tabla de arriba.

Pero antes pasemos al mínimo común múltiplo (m.c.m.):

El m.c.m. de dos números Z1 y Z2 se obtiene dividiendo el producto de ambos entre el M.C.D.

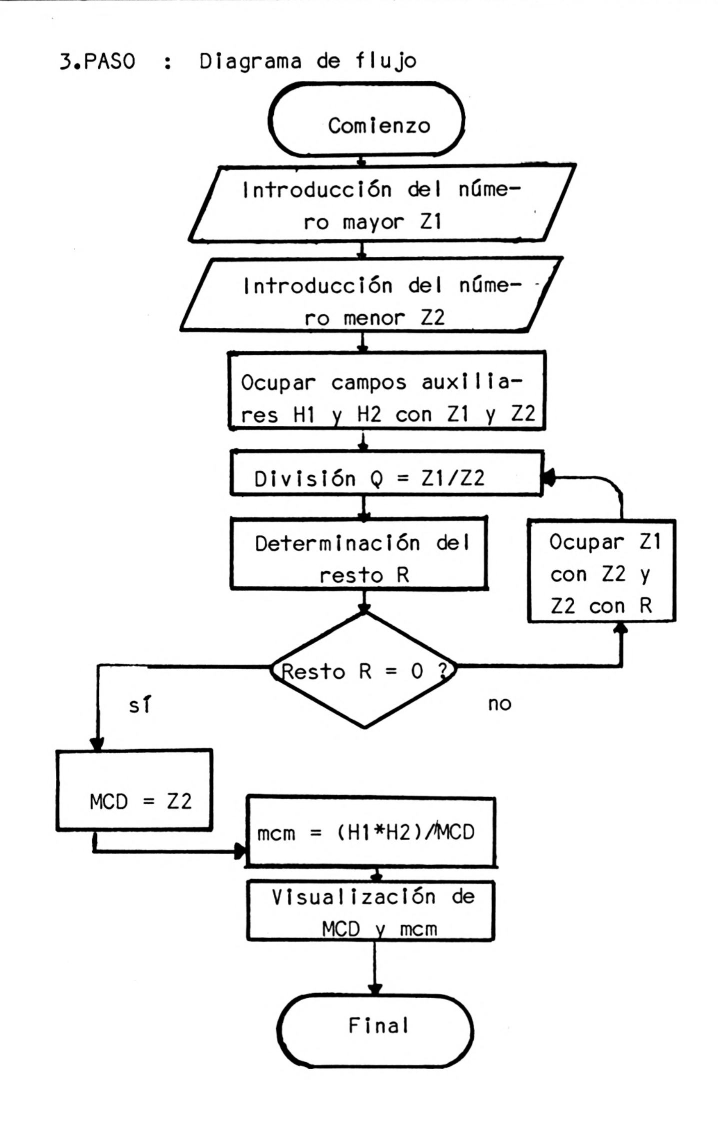
Así, para los números Z1=78 y Z2=42 se obtiene:

$$m.c.m. = \frac{Z1 * Z2}{M.C.D.} = \frac{78 * 42}{6} = 546$$

Para efectuar este cálculo dentro de un programa BASIC según la tabla de arriba cambiando contínuamente el valor

que ocupa cada campo, calcular el M.C.D. e introducir los valores Z1 y Z2 en dos campos auxiliares H1 y H2 de ''memoria intermedia'' para que estén disponibles para el cálculo del m.c.m.

3.PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM M2 - MCD Y MCM
20 CLS
30 PRINT "PROGRAMA PARA DETERMINAR EL M.C.D. Y"
40 PRINT "M.C.M DE LOS DOS NUMEROS INTRODUCIDOS"
50 PRINT: PRINT: PRINT TAB(10) "PROF. DR. W. VOSS, 1984": PRIN
60 INPUT "NUMERO MAYOR      : "; Z1
70 PRINT: PRINT
80 INPUT "NUMERO MENOR      : "; Z2
90 H1 = Z1: H2 = Z2
100 Q = Z1 / Z2
110 R = Z1 - Z2 * INT(Q)
120 IF R > 0 THEN Z1 = Z2 : Z2 = R : GOTO 100
130 MCD = Z2
140 MCM = (H1 * H2) / MCD
150 PRINT: PRINT: PRINT "MCD = "; MCD
160 PRINT: PRINT: PRINT "MCM = "; MCM
170 PRINT: PRINT "FINAL DEL CALCULO" : END
```

5. PASO : LISTA DE VARIABLES

MCD = Máximo común divisor
H1 = Campo auxiliar 1
H2 = campo auxiliar 2
MCM = Mínimo común mítiplo.
B = cociente de los números Z1 y Z2.
R = Resto de la división
Z1 = Número mayor
Z2 = Número menor

6. PASO : Descripción del programa

Línea 10-50 : Título y explicaciones

Línea 60 : Introducción de un número

Línea 70 : Dos líneas en blanco

Línea 80 : Introducción del otro número

Línea 90 : Ocupación de los dos campos auxiliares H1 y H2 con los valores introducidos, para más tarde seguir disponiendo de los mismos (ver 140)

Línea 100 : Determinación del cociente de los dos números introducidos.

Línea 110 : Determinación del resto de la división utilizando la función INT

Línea 120 : Si este resto es mayor que 0 se ocupa el campo Z1 con el segundo número Z2 y el campo Z2 con el resto R, volviendo el programa a la línea 100

Línea 130 : Si, en cambio, el resto es igual a cero, en el campo Z2 tendremos el M.C.D., que será transferido al campo M.C.D.

Línea 140 : De acuerdo con nuestra reflexión,el mcm resulta del producto de ambos números (que aún se encuentran en los campos H1 y H2), dividido entre el MCD.

Línea 150-160 : Impresión de los resultados con una separación de algunas líneas en blanco

Línea 170 : Finalización del programa

7. PASO : Resultados del programa

Después del comando RUN, el programa indica, en una ''pantalla limpia'', lo siguiente :

```
PROGRAMA PARA DETERMINAR EL M.C.D.
Y EL M.C.M. DE LOS DOS NUMEROS
     INTRODUCIDOS

    PROF. DR.W.VOSS, 1984

NUMERO MAYOR  : ?
```

Cuando aparezca el interrogante, que como ya se sabe, indica una interrupción del programa, introducir por

ejemplo el valor 78; a continuación el programa pedirá el segundo número:

NUMERO MENOR : ?

Introduzcamos por ejemplo el número 42 y el ordenador llevará a pantalla los resultados siguientes:

M.C.D. = 6

M.C.M. = 546

FINAL DEL CALCULO

2.4 Comprobación de números primos

1. PASO : Presentación del programa

Se trata de desarrollar un programa que compruebe si es primo o no un número entero positivo cualquiera que hemos introducido.

Como ya es sabido, los números primos sólo son divisibles por 1 y por sí mismos.

2. PASO : Análisis del problema

Para comprobar si un número dado es primo, hay que dividirlo entre todos los divisores mayores que 1, pero menores que él mismo. Mientras aparezca un resto en estas divisiones o el cociente no sea entero, habrá que seguir dividiendo por otro divisor.

De obtener en una de estas divisiones un resultado sin resto, el número a examen no será primo.

Si, en cambio, se obtiene un resto en todas las divisiones posibles, se llegará a la conclusión de que el número inicial es primo.

Los divisores que deben ser probados comienzan por el 2 y pueden finalizar a la mitad del número inicial; los divisores mayores que éste no es necesario probarlos, porque en ningún caso darán un cociente entero.

Ejemplo : Número inicial X = 25

```
1. Divisor : 2    25/2 = 12 resto 1
2. Divisor : 3    25/3 =  8 resto 1
3. Divisor : 4    25/4 =  6 resto 1
4. Divisor : 5    25/5 =  5 resto 0
                                  ===
```

25 no es ningún número primo.

Ejemplo : Número de partida X = 7

1. Divisor : 2 7/2 = 3 resto 1
2. Divisor : 3 7/3 = 2 resto 1

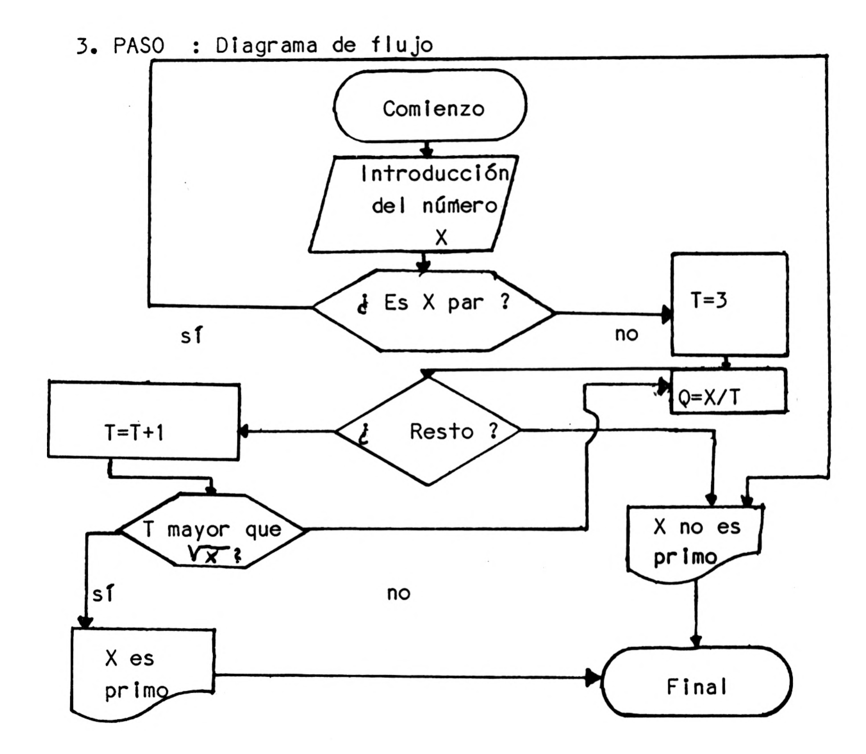
No es necesario probar más divisores, porque el resultado de la división ya no puede ser entero.

Como en este ejemplo no aparece ningún resto 0, podemos decir que:

7 es un número primo.

Estos ejemplos muestran que hay que aumentar paso a paso el divisor comenzando por el valor 2 hasta alcanzar como máximo el valor X/2.

3. PASO : Diagrama de flujo



Este ejemplo muestra como puede simplificarse de modo considerable - y con ello acelerarse - el programa si analizamos cuidadosamente el problema:

Un número par nunca puede ser primo pues, en todo caso, éste ya es divisible por 2. Por ello pueden excluirse los números pares como "números no - primos" desde el comienzo del programa.

Nos quedan los números impares. Para un número cualquiera X no es necesario que el divisor llegue a ser X/2; basta con que probemos divisiones hasta máximo $\sqrt{X}$. Si hasta este punto no es posible ninguna división sin resto, tampoco lo será con divisores mayores.

Invitamos al lector a que intente demostrar esta afirmación.

4. PASO : Programa

```
10 REM M3-NUMERO PRIMO
20 CLS
30 PRINT "PROGRAMA PARA COMPROBAR SI EL NUMERO INTRODUCIDO"
40 PRINT "ES O NO UN NUMERO PRIMO"
50 PRINT: PRINT: PRINT TAB(10) "PROF. DR. W. VOSS, 1984": PRINT
60 INPUT "UN NUMERO ENTERO POR FAVOR : "; X
70 T = 2
80 IF X / 2 = INT(X / 2) THEN 140
90 Q = X / T
100 IF Q = INT(Q) THEN 140
110 T = T + 1
120 IF T < = SQR(X) THEN 90
130 PRINT: PRINT: PRINT X; "ES UN NUMERO PRIMO": GOTO 150
140 PRINT: PRINT: PRINT X; "NO ES UN NUMERO PRIMO, SINO "
145 PRINT "QUE ES DIVISIBLE ENTRE "; T
150 PRINT: PRINT: PRINT "FIN DEL CALCULO": END
```

5. PASO Lista de variables

Q = Cociente entre el número X y el divisor T

T = Divisor

X = Número a introducir para comprobar si es primo o no.

6. PASO : Descripción del programa

Línea 10-50 : Título, limpieza de la pantalla e impresión del título.

Línea 60 : Demanda del número a comprobar.

Línea 70 : Ocupación del campo divisor con el 2.

Línea 80 : Si X es divisible por 2 (entonces X/2 = INT (X/2)), pasar a la línea 140; si no es así,proseguir en la línea 90.

Línea 90 : División de X entre T (se obtiene Q).

Línea 100 : Si en una de las divisiones el resto es 0 (Q = INT(Q)), seguir en la línea 140, en caso contrario seguir en la línea 110.

Línea 110 : Aumentar el divisor T en 1.

Línea 120 : Mientras T siga siendo más pequeño que $\sqrt{X}$,volver a la línea 90, de lo contrario proseguir en 130.

Línea 130 : Si se alcanza esta línea es porque ninguna de las divisiones ha dado resto 0.

Esto significa que X debe ser número primo.

Se imprime el resultado y tiene lugar un salto a la línea 150.

Línea 140 y 145 : La línea 140 se alcanza únicamente en el caso de haber obtenido resto 0 en una de las divisiones.

Esto supone que X no es primo, y que es (como mínimo) divisible por el valor que hay en el campo T.

Esto se visualiza.

Línea 150 : Finalización del programa.

7. PASO : Resultados

En este y en próximos ejemplos prescindiremos de repetir la impresión del título.

Después de aparecer el título, el programa pide el número a comprobar y lo almacena en el campo X (también prescindiremos en este y en próximos ejemplos de reproducir los correspondientes textos de demanda).

Introduzcamos p.e. el número 13 y el ordenador contestará:

```
13 ES UN NUMERO PRIMO

FIN DEL CALCULO
```

Sí en cambio introducimos el número 27, el ordenador nos contestará:

```
27 NO ES UN NUMERO PRIMO, SINO
                         QUE ES DIVISIBLE POR 3


FIN DEL CALCULO
```

2.5 Ecuación de segundo grado

1. PASO : Presentación del problema

La resolución de ecuaciones de segundo grado se cuenta entre los problemas fundamentales de la matemática en los últimos niveles de EGB y primeros del BUP y prepara para problemas de mayor complejidad.

La forma general de una ecuación de segundo grado en la cual hay que despejar la incógnita x es:

$$(1) \qquad ax^2 + bx + c = 0$$

Dividiendo esta igualdad por el coeficiente a, se obtiene:

$$(2) \qquad x^2 + px + q = 0$$

Se obtienen dos soluciones x y x a partir de la fórmula siguiente:

$$(3) \qquad x_{1/2} = -p/2 \pm \sqrt{p^2/4 - q}$$

Según la constelación de coeficientes a, b y c, la fórmula (3) nos llevará a resultados distintos.

Por ejemplo, cuando el valor bajo la raíz es negativo, obtendremos resultados fuera del campo de los números reales. Este valor se llama discriminante(D).

Se obtienen también soluciones especiales cuando el coeficiente a es igual a 0;en este caso ya no tenemos una ecuación de segundo grado.

Por ello, de ahora en adelante contemplaremos sólo los casos ''auténticos'' (a $\neq$ 0) y al mismo tiempo con soluciones reales (D>0). En todos los demás casos el programa deberá generar un mensaje de interrupción.

Para resolver el problema utilizaremos la fórmula (3) en su variante (4):

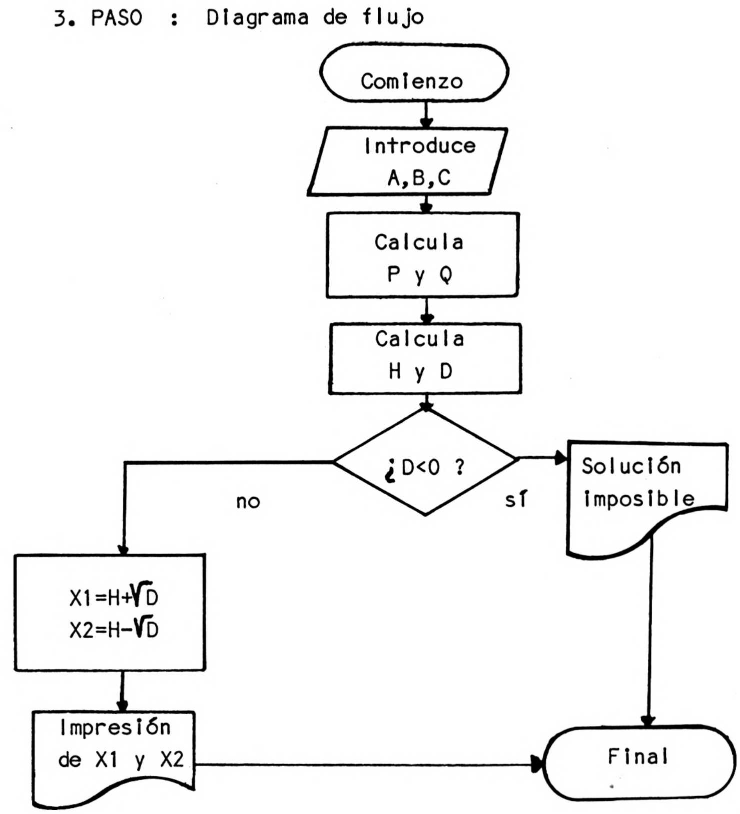
$$(3) \qquad x_{1/2} = -p/2 \pm \sqrt{p^2/4 - q}$$

$$(4) \qquad x_{1/2} = H \pm \sqrt{D}$$

H es un valor auxiliar que equivale a -p/2

El correspondiente programa BASIC debe determinar primero los valores H y D, para luego imprimir las soluciones o un mensaje de interrupción según sea D>0 o D<0.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM M4-ECUACION DE SEGUNDO GRADO
20 CLS
30 PRINT "PROGRAMA PARA RESOLVER UNA "
40 PRINT "ECUACION DE SEGUNDO GRADO "
50 PRINT: PRINT: PRINT TAB(10) "PROF. DR. W. VOSS, 1984": PRINT
60 INPUT "COEFICIENTE DEL TERMINO CUADRATICO A : "; A
70 INPUT "COEFICIENTE DEL TERMINO LINEAL B     : "; B
80 INPUT "TERMINO INDEPENDIENTE C              : "; C
90 P = B/A: Q = C/A
100 H = -P / 2 :D = P * P / 4 - Q
110 IF D < 0 THEN PRINT: PRINT "SOLUCION IMPOSIBLE": GOTO 150
120 X1 = H + SQR(D): X2 = H - SQR(D)
130 PRINT: PRINT: PRINT "SOLUCION PRIMERA   X1 = "; X1
140 PRINT: PRINT: PRINT "SOLUCION SEGUNDA   X2 = "; X2
150 PRINT: PRINT: PRINT "FIN DEL CALCULO": END
```

5. PASO : Lista de variables

A = Coeficiente de la parte cuadrática
B = Coeficiente de la parte lineal
C = Constante en la ecuación cuadrática

D = Discriminante
H = -P/2
P = B/2
Q = C/A

X1 = Primera solución
X2 = Segunda solución

6. PASO : Descripción del programa

Línea 10-50 : Comentario e impresión del título

Línea 60-80 : Introducción de los coeficientes de la igualdad cuadrática (en la forma (1))

Línea 90 : Determinación de P y de Q según la igualdad (2)

Línea 100 : Determinación del valor auxiliar H y del discriminante D

Línea 110 : Si D es menor que 0, tiene lugar un mensaje de interrupción y se pasa a la línea 150

Línea 120 : Si, en cambio, el discriminante no es negativo, se calculan las dos soluciones X1 y X2

Línea 130-140: Ambas soluciones se visualizan en la pantalla

Línea 150 : Final del programa

7. PASO : Resultados

Introduzcamos p.e.:

A = 4,
B = 4,
C = -24,

¡El lector debería probar también otros valores!

Se trata de la ecuación de segundo grado:

$4x^2 + 4x - 24 = 0$,

tras lo cual el programa visualiza los siguientes resultados:

```
X1 = 2
X2 = -3
FINAL DE LOS CALCULOS
```

2.6 Número e

1. PASO : Presentación del problema

El número de Euler, abreviado e, llamado también constante de crecimiento, juega un papel importante en el desarrollo matemático de series, que pueden ser contempladas como cadenas de crecimiento o de desarrollo.

Este sirve además como base de los llamados logarítmos neperianos, que volveremos a encontrarnos en el capítulo dedicado a problemas de economía.

Si nos hemos olvidado del valor del número e, podemos calcularlo con ayuda de un programa BASIC adecuado y siempre sabiendo cómo surge e.

Se trata del límite de la siguiente sucesión, cuando n tiende hacia infinito:

$$e = (1 + 1/n)^n \quad \text{para n hacia infinito.}$$

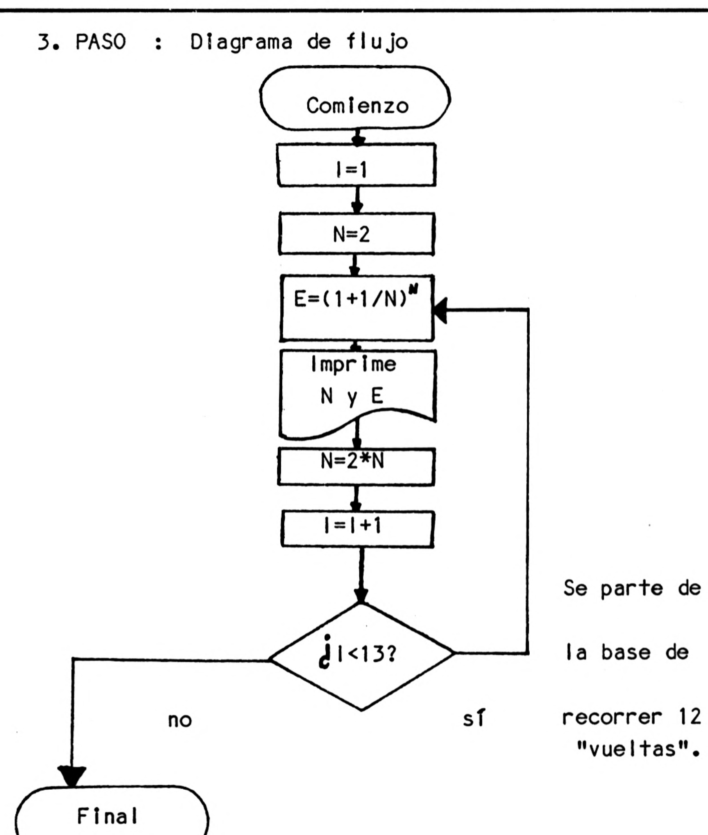
2. PASO : Análisis del problema

El análisis del problema es muy sencillo en este caso, dado que se trata únicamente de programar la fórmula anterior.

Este programa puede probarse con valores crecientes y alternativos de n. Como el valor e se alcanza sólo con n tendiendo hacia infinito, recomendamos hacer crecer rápidamente n, por ejemplo doblando su valor en cada cálculo.

Ello exige, no obstante, empezar con un valor de n mayor que 1 (es decir, p.e. n = 2), para que n no se quede estancado en 1 al duplicarlo.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM M5-EULER
20 CLS
30 PRINT "PROGRAMA PARA DETERMINAR EL NUMERO E"
40 PRINT
50 PRINT: PRINT: PRINT TAB(9) "PROF. DR. W. VOSS, 1984": PRINT
60 N = 2
70 PRINT "    N","    E": PRINT
80 FOR I = 1 TO 12
90 E = (1 + 1 / N) ^ N
100 PRINT N,E
110 N = N * 2
120 NEXT I
130 PRINT "FINAL": END
```

5. PASO : Lista de variables

E = Número e (valor aproximado)

I = Indice de conteo de bucles

N = Parámetro en la fórmula para el cálculo del número e

6. PASO : Descripción del programa

Línea 10-50 : Título

Línea 60 : Ocupación del parámetro N con el valor inicial 2

Línea 70 : Impresión del título de la tabla y de una línea en blanco

Línea 80-120 : Recorrer 12 veces el bucle. En cada vuelta se calcula E, se imprimen N y E y se duplica N

Línea 130 : Final del programa

7. PASO : Resultados

Este programa genera una tabla, en la que se enfrentan los valores crecientes de N con los valores resultantes de E.

En esta tabla puede observarse como e se acerca a un valor fijo.

Para acercarnos aún más al límite, deberíamos recorrer más vueltas de bucle.

Entonces habría que modificar la línea 80, por ejemplo del modo siguiente:

```
80 FOR I = 1 TO 30
```

En todo caso, hay que tener en cuenta que los valores demasiado altos de n pueden conducir a errores, debido a las imprecisiones de cálculo del ordenador.

La tabla resultante del programa anterior ofrece este aspecto:

```
   N             E

  2           2.24999997
  4           2.44140621
  8           2.56578451
  16          2.63792848
  32          2.67699016
  64          2.69734499
  128         2.70773919
  256         2.71299187
  512         2.71562546
  1024        2.71695025
  2048        2.72761466
  4096        2.71794529
FINAL
```

2.7 Cálculo de porcentajes

1. PASO : Presentación del problema

La regla de tres juega un papel fundamental en el cálculo de porcentajes. Por ejemplo responde a la pregunta de qué parte de una cantidad total G representa un determinado importe B. Podemos ilustrar este problema con un ejemplo típico:

El importe de una factura asciende a Ptas. 45000.--. El cliente debe pagar adicionalmente el llamado impuesto sobre el valor añadido, que supone un 10% del importe neto de la factura, de lo cual resulta:

Importe bruto = Importe neto + Impuesto

El problema que aquí se plantea consiste entonces en determinar, primero el importe del impuesto y después el importe bruto, para un importe neto y un porcentaje del impuesto dados.

Naturalmente podría pedirse el importe neto sabiendo el porcentaje y el importe bruto.

2. PASO : Análisis del problema

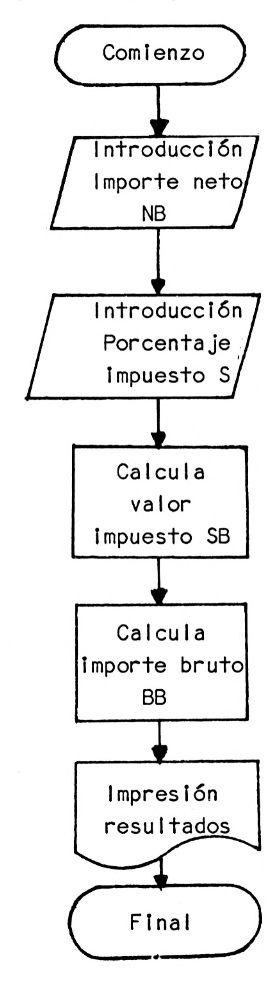
El ejemplo citado arriba puede solucionarse por regla de tres del modo siguiente (señalándonos así el modo de resolución):

```
Importe neto = 100 % equivalente a 45000.- Ptas.
Impuesto     = 14 % equivalente a    x     Ptas.
-----------------------------------------------

     45000*10
x =  --------  =  4500.- (Ptas.)
       100
```

Con ello queda claro el camino a seguir: Se trata de introducir en un programa general el importe neto y el porcentaje del impuesto (o de forma más general, el porcentaje, que puede elegirse arbitrariamente para cualquier aplicación).

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM M6-PORCENTAJE
20 CLS
30 PRINT "PROGRAMA PARA ESTIMACION DE PORCENTAJES"
40 PRINT
50 PRINT: PRINT: PRINT TAB(9) "PROF. DR. W. VOSS, 1984": PRINT
60 INPUT "SUMA NETA              : "; NB
70 PRINT: INPUT "PORCENTAJE   : "; SS
80 SB = NB * SS / 100
90 BB = NB + SB
100 PRINT: PRINT: PRINT "SUMA NETA                    : "; NB
110        PRINT: PRINT "PORCENTAJE CALCULADO    : "; SB
120        PRINT: PRINT "BRUTO CALCULADO         : "; BB
130 PRINT: PRINT "FINAL": END
```

5. PASO Lista de variables

BB = Importe bruto
NB = Importe neto
SB = Valor porcentual
SS = Porcentaje

6. PASO : Descripción del programa

Línea 10-50 : Título.

Línea 60-70 : Introducción del importe neto y del porcentaje.

Línea 80-90 : Cálculo del importe del porcentaje y del importe bruto.

Línea 95 : Cuatro líneas en blanco.

Línea 100-120 : Impresión en pantalla de los resultados

Línea 130 : Final del programa.

7. PASO : Resultados

Si se introduce por ejemplo el importe neto 200 y el porcentaje 15%, el programa contestará:

```
IMPORTE NETO       :  200
VALOR PORCENTUAL   :  30
IMPORTE BRUTO      :  230
FIN
```

2.8 Dado

1. PASO : Presentación del problema

Presentamos un programa que "echa" 100 veces los dados y que calcula el promedio de puntos obtenido.

Para este fin puede hacerse uso de la función RND (RND = random = azar), que genera números aleatorios entre el 0 y el 1.

Números aleatorios son aquéllos que aparecen de forma irregular y no sistemática.

2. PASO : Análisis del problema

En el AMSTRAND CPC 464/6128 puede utilizarse la función RND sin parámetros.

El lector debería probar simplemente esta función, introduciendo por ejemplo:

```
PRINT RND
```

y seguidamente, una vez más:

```
PRINT RND
```

o bien sería aún mejor introducir:

```
FOR I = 1 TO 10 PRINT RND : NEXT
```

pulsando después 'RETURN'.

En la pantalla aparecerán números aleatorios entre 0 y 1.

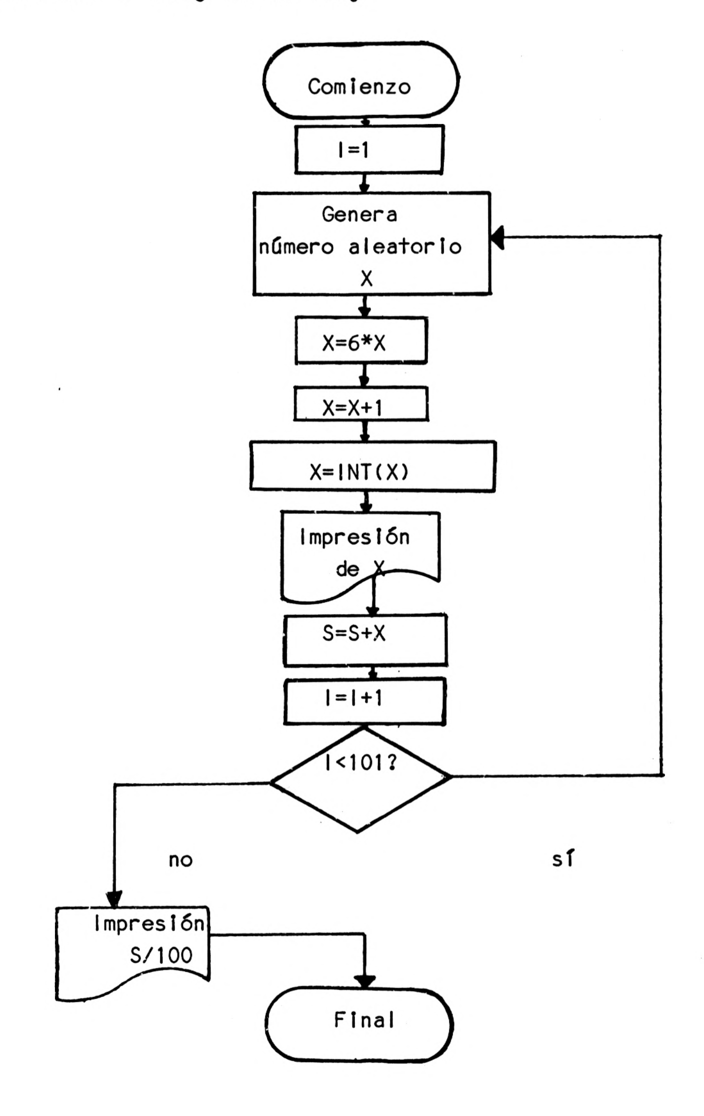
No obstante queremos obtener como resultado 100 valores enteros situados entre 1 y 6, como es el caso cuando se echa un dado.

El esquema siguiente ilustra cómo pueden generarse números enteros entre el 1 y el 6 a partir de números aleatorios tomados del ámbito de valores entre 0 y 1:

Elaboración	Resultado
Función RND	Entre 0 y menos de 1
* 6	entre 0 y menos de 6
+ 1	entre 1 y menos de 7
Función INT	entero entre 1 y 6

Con ello se ha trazado ya el camino para resolver el problema. Como finalmente se desea obtener el promedio, habrá que sumar todos los resultados y dividir esta suma entre 100.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM M7-DADO
20 CLS
30 PRINT "PROGRAMA PARA GENERAR 100 TIRADAS"
40 PRINT "DE DADO Y CALCULAR EL PROMEDIO DE"
50 PRINT TAB(16) "PUNTOS."
60 PRINT: PRINT: PRINT
70 PRINT TAB(9) "PROF. DR. W. VOSS, 1984": PRINT
80 I = 1
90 X = RND
100 X = X * 6: X = X + 1: X = INT(X)
110 PRINT X:
120 S = S + X
130 I = I + 1
140 IF I < 101 THEN 90
150 AM = S / 100
160 PRINT: PRINT: PRINT
170 PRINT: PRINT "VALOR MEDIO  =  "; AM
180 PRINT: PRINT "FINAL DE LA IMPRESION": END
```

5. PASO Lista de variables

AM = Valor medio
I = Indice variable
S = Suma de todas las tiradas
X = Valor aleatorio

6. PASO : Descripción del programa

Línea 10-70 : Título

Línea 80 : Ajuste del contador de vueltas

Línea 90-100 : Generación de un número aleatorio y transformación en un número entero situado entre 1 y 6

Línea 110 : Impresión

Línea 120 : Incremento de la suma S en el resultado de la tirada X

Línea 130 : Salto de uno del contador de vueltas

Línea 140 : Mientras I sea menor que 101 vuelta a la línea 90

Línea 150-170 : Cálculo del promedio e impresión

Línea 180 : Final del programa

7. PASO : Resultados

Después de iniciar el programa, éste genera tras el título 100 sucesos aleatorios de tirada de un dado y los imprime línea a línea (pensar por qué tiene lugar una impresión línea a línea).
A continuación el ordenador calcula, a partir de los 100 sucesos, el promedio y lo visualiza. Si el generador de números aleatorios funciona correctamente, el resultado deberá acercarse a 3.5.

Como los resultados dependen del azar, lógicamente no los podemos representar aquí.

2.9 Exactitud de los cálculos

Un ordenador como el AMSTRAND CPC 464/6128 reserva un número limitado de bytes para cada número real con el que va a calcular. Esto significa que la precisión con la que trabaja está limitada.

Puede verse esto claramente si introducimos valores "grandes" en el programa que sigue, haciendo que se impriman a continuación:

```
10 REM EXACTITUD DE CALCULO
20 A = 12345678
30 B = 123456789
40 C = 1234567891
50 D = 12345678912
60 ?A: ?B: ?C: ?D
70 END
```

Obtenemos los siguientes resultados:

```
 12345678
 123456789
 1.234567E+09
 1.234567E+10
```

Es decir, si el ordenador recibe números de más de 10 cifras (ejemplo D) a la hora de imprimir, los pasa a la llamada representación exponencial, como por ejemplo para C:

1.234567E+09

Esto debe leerse como:

1.234567 * 10^9 = 1.234567 * 1 000 000 000

Este producto da como resultado

1 234 567 000

El ordenador ha redondeado por debajo la última posición del valor de C, que ya no puede considerar.

En todo caso, como puede mostrarse por medio de otros ejemplos, este redondeo no se efectúa simplemente "recortando" la posición.

El principal problema que surge aquí es el siguiente:

Si, por ejemplo, sumamos 1 al valor D en el programa anterior, dentro de una nueva línea de programa 55:

```
55 E = D + 1
```

y hacemos imprimir también el valor E:

```
65 PRINT E
```

no advertiremos ninguna diferencia entre los valores de D y E -¿el ordenador calcula mal!

Este problema debe ser considerado siempre que se deba operar números muy grandes con números muy pequeños (o viceversa).

Capítulo 3 : Química

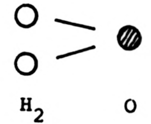
3.1 Otras instrucciones BASIC

Para confeccionar programas en el campo de la química pueden aprobecharse magníficamente las posibilidades gráficas y de color del AMSTRAND CPC 464/6128 -por ejemplo para representar moléculas o enlaces químicos.

Tomemos el caso de la molécula de agua:

H_2O

que, como ya es sabido, está compuesta de dos átomos de hidrógeno (H) y de un átomo de oxígeno (O), que se representan de la forma siguiente:



Para generar representaciones como ésta se necesitan algunas instrucciones adicionales. Estas se refieren a la utilización de caracteres especiales (en un próximo capítulo hablaremos de las posibilidades de los llamados gráficos de alta resolución).

Para confeccionar este tipo de dibujos resulta muy adecuado el juego de caracteres gráficos del AMSTRAND CPC 464 (ver manual de instrucciones).

Estos caracteres, que pueden hallarse bajo determinados códigos ASCII (para ello resultan necesarios algunos caracteres especiales adicionales) pueden seleccionarse con la función que anteriormente ya habíamos nombrado

CHR$ (argumento)

El argumento de esta función es el número de código ASCII de símbolo deseado.

Si el usuario introduce por ejemplo:

```
PRINT CHR$(231)
```

se visualiza en la pantalla un símbolo semejante a una bolita.

A continuación les presentamos algunas indicaciones para la composición de colores para aquellos que tengan conectado un monitor o televisor de color (de todas formas, todos los programas que trabajan con efectos de colores también funcionan con televisores en blanco y negro, donde los diferentes colores se representan por diferentes tonalidades de verde o sombreados).

Al conectar el CPC 464/6128 éste se encuentra en modalidad 1 (40 caracteres por línea) y en un monitor de color conectado se visualiza todo en amarillo sobre fondo azul oscuro.

Los colores de emisión pueden alterarse con la instrucción siguiente:

Sentencia 12:

```
nn PEN número
```

En modalidad 1, se pueden representar 4 colores diferentes con ésta instrucción, según los siguientes parámetros:

Número	Color
1	Amarillo
2	Azul claro
3	Rojo
4	Azul

El último color mencionado provoca una salida que no es visible en el fondo azul. Con el siguiente programa se pueden visualizar 3 bolitas de diferentes colores, una debajo la otra:

```
10 FOR I = 1 TO 3
20 PEN I
30 PRINT I,CHR$(231)
40 NEXT I
50 END
```

Advertencia:

Si al experimentar con los colores (o de cualquier otra forma) llega a un punto muerto, donde ya no funciona nada, se puede reinicializar el ordenador pulsando simultáneamente las siguientes teclas:

CTRL SHIFT y ESC

Sin embargo se pierde el programa que actualmente resida en el ordenador.
Además necesitamos en éste capítulo y en los subsiguientes instrucciones que nos permitan introducir información al ordenador de forma más ágil que la que resulta utilizando las instrucciones LET e INPUT.
Para ello se ofrece la instrucción DATA en combinación con READ.

Instrucción 13:
----

nn DATA Valor 1, Valor 2, Valor 3, ...

Instrucción 14:
----

nn READ variable 1, variable 2, variable 3,....

Ejemplo:

```
10 REM INTRODUCCION DE DATOS
20 DATA 4600,DORTMUND,4630,PARIS,8000
30 DATA MUNICH,2000,HAMBURGO
40 READ P1,01$,P2,02$,P3,03$,P4,04$
50 ?P1,01$:P2,02$:?P3,03$:?P4,04$
60 END
```

Invitamos al lector a probar este programa.

Cuando la cantidad de datos sea elevada, recomendamos utilizar un único nombre de variable que, no obstante, tomará varios valores. Esto se consigue utilizando variables indexadas como las que ya conocemos de las matemáticas. Normalmente en términos matemáticos se escribe:

x_1, x_2, x_3, ... término general x_i

Esto puede llevarse a cabo de la misma forma con el lenguaje BASIC:

X(1), X(2), X(3), ... término general X(I)

(en general:
nombre variable (nombre del índice))

Cuando utilizamos estas variables indexadas hay que comunicar al ordenador al comienzo del programa cuántas posiciones de memoria deben mantenerse libres para cada una de las variables en el principio del programa.

Para ello utilizaremos la instrucción DIM:

Instrucción 15:

```
nn DIM nombre var. 1 (num1), nombre var.2 (num2), ...
```

Así por ejemplo la instrucción:

```
10 DIM X(20)
```

mantiene libres 21 posiciones de memoria para la variable X.

Aquí se trata de las posiciones X(0), X(1), X(2), X(3),..., X(20).

Para completar esta explicación añadiremos que son también posibles indexados dobles:

La instrucción:

```
10 DIM Z(2,3)
```

mantiene libres 3*4 posiciones de memoria para la variable Z, que uno puede imaginarse divididas en tres líneas y cuatro columnas:

Z(0,0)	Z(0,1)	Z(0,2)	Z(0,3)
Z(1,0)	Z(1,1)	Z(1,2)	Z(1,3)
Z(2,0)	Z(2,1)	Z(2,2)	Z(2,3)

El formato general es en este caso:

Z(I,J)

El primer índice (I) es el llamado índice de línea y el segundo índice (J) es el llamado índice de columna.

Para finalizar queremos indicar que la instrucción READ de un programa, toma valores de los DATAS en orden sucesivo. Ello significa que una segunda instrucción READ proseguirá con la lectura de los DATAS, allí donde dejó de hacerlo la última instrucción READ.

Aún así, a menudo es más razonable volver a leer desde el principio la lista de DATAS con una segunda (o siguiente) instrucción READ.

Para hacer posible esto necesitamos una instrucción nueva:

Instrucción 16:

```
nn RESTORE
```

Esta instrucción provoca que el ''cursor de lectura'' sea emplazado nuevamente al comienzo de la primera instrucción DATA, de modo que los datos sean leídos nuevamente desde el principio con la siguiente instrucción READ.

3.2 La molécula de agua

1. PASO : Presentación del problema

Vamos a generar, con ayuda de un programa BASIC, la representación gráfica de la molécula de agua (H_2O); véase también el punto 3.1. Los átomos de hidrógeno se presentarán p.e. en forma de bolitas amarillas y el átomo de oxígeno como una bolita roja.

Así, este programa constituye la base de programas más complejos de este tipo de representación gráfica.

2. PASO : Análisis del problema

Una vez tenemos ante nosotros el gráfico que debe generarse con este programa, el análisis del problema resulta muy sencillo:

Se trata de producir exactamente el mismo dibujo que ya realizamos al comienzo del punto 3.1.

Observaremos que este dibujo se compone de cuatro líneas:

1. Línea : Una bolita amarilla y una línea oblicua desde extremo superior izquierdo hacia extremo inferior derecho.

2. Línea : Una bolita roja (desplazada hacia la derecha)

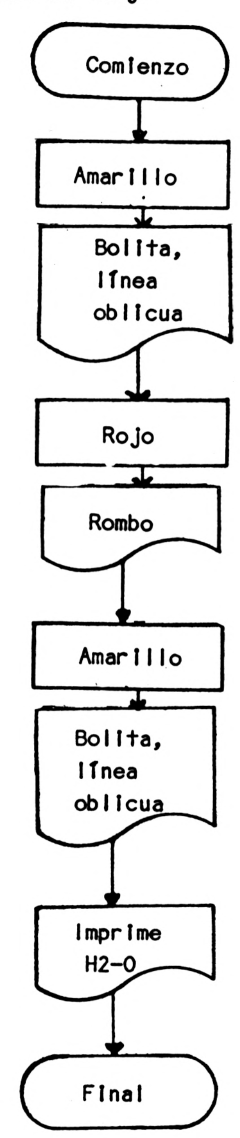
3. Línea : Una bolita amarilla situada debajo de la primera línea y una raya oblicua desde el extremo inferior izquierdo hacia el extremo superior derecho.

4. Línea : texto : H2-O.

La bolita podemos generarla, como ya hicimos antes, con el carácter ASCII número 231 (con ayuda de la función CHR$).

De la misma forma, la línea oblicua con dirección desde extremo superior izquierdo hacia extremo inferior derecho (\) tiene el código 205 y la línea inversa (/) el código 204.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM Q1-H2O
20 CLS
30 PRINT "PROGRAMA PARA LA REPRESENTACION GRAFICA"
40 PRINT "DE LA MOLECULA DE AGUA H2-O"
50 PRINT: PRINT : PRINT TAB(10) "PROF. DR. W. VOSS, 1984": PRINT
60 PRINT: PRINT: PRINT
70 PEN 1: PRINT TAB(10) CHR$(231);
80 PRINT TAB(12) CHR$(205)
90 PEN 3: PRINT TAB(14) CHR$(231)
100 PEN 1: PRINT TAB(10) CHR$(231);
110 PRINT TAB(12) CHR$(204)
130 PRINT: PRINT TAB(10) "H2 - O"
140 PRINT: PRINT: PRINT "FINAL": END
```

5. PASO : Lista de Variables

En este programa no se utiliza ninguna variable

6. PASO : Descripción del programa

Línea 10 - 60	Título
Línea 70	Cambiar a color amarillo y visualizar bolita en columna 10; supresión de avance de línea
Línea 80	Visualización de una línea oblicua en columna 12
Línea 90	Cambiar color a rojo y visualizar bolita en columna 14
Línea 100	Cambiar color a amarillo y visualizar bolita en columna 10; supresión de avance de línea.
Línea 110	Visualización de una línea oblicua en columna 12
Línea 130	Visualizar H2-O a partir de columna 10 después de 2 líneas en blanco.
Línea 140	Final del programa

7. PASO : Resultados

La impresión en pantalla de este programa es similar a la que presentamos en el punto 3.1.

3.3 Reacción química

1. PASO : Presentación del problema

Las reacciones químicas también pueden representarse gráficamente de una forma clara, como se demuestra en el ejemplo del proceso de fabricación del acero.

La fabricación del acero aparece en los países industrializados con la producción de hierro bruto en los altos hornos. Esta producción se desarrolla en los siguientes pasos:

1. Paso : Se quema aire caliente con carbón para obtener monóxido de carbono:

$$2\,C + O_2 \longrightarrow 2\,CO$$

carbón oxígeno monóxido de carbono

2. Paso : El monóxido de carbono (CO) caliente asciende por el horno y reacciona allí con el óxido férrico (Fe O):

$$Fe_2O_3 + 3\ CO \longrightarrow 2\ Fe + 3\ CO_2$$

Oxido férrico	Monóxido de carbono	Hierro	Dióxido de carbono

De este modo se obtiene hierro bruto para su posterior utilización.

Presentamos ahora un programa BASIC que sirve para mostrar estas reacciones químicas.

Para la representación de los átomos y de las moléculas nos serviremos de las mismas posibilidades gráficas que en el ejemplo precedente.

2. PASO : Análisis del problema

En realidad, con la descripción anterior y considerando lo dicho en el ejemplo anterior, el análisis del problema ya queda resuelto, de modo que podemos ahorrarnos cualquier otra explicación.

3. PASO : Diagrama de flujo

Tampoco aquí resulta necesario el diagrama de flujo, por los mismos motivos que en el punto anterior.

4. PASO : Programa

```
10 REM Q2-REAC. QUIMICA
20 CLS
30 PRINT "PROGRAMA PARA REPRESENTAR UNA REACCION"
40 PRINT TAB(13) "QUIMICA"
50 PRINT: PRINT : PRINT TAB(10) "PROF. DR. W. VOSS, 1984": PRINT
60 PRINT: PRINT: PRINT
70 PEN 1: PRINT TAB(10) CHR$(231);
75 PRINT TAB(25) CHR$(231); : PEN 2: PRINT CHR$(231)
80 PRINT:  PRINT TAB(12) "+  ";:PEN 2:  PRINT  CHR$(231);:  PRINT
   CHR$(231);
85 PRINT "   =   "
86 PRINT
90 PEN 1: PRINT TAB(10) CHR$(231);
100 PRINT TAB(25) CHR$(231);:PEN 2: PRINT CHR$(231)
110 PRINT: PRINT
120 PEN 1: PRINT TAB(10) "2C + 0    =    2CO"
130 PRINT TAB(16) "2"
140 PRINT: PRINT: PRINT: PRINT "POR FAVOR INTRODUZCA CONT": STOP
```

```
150 CLS
160 PEN 2: PRINT TAB(33) CHR$(231)
170 PEN 1: PRINT TAB(18) CHR$(231);: PEN 2: PRINT CHR$(231);
180 PEN 1: PRINT TAB(32) CHR$(231)
190 PEN 2: PRINT  TAB(12)  CHR$(231);:  PEN  3:  PRINT  TAB(26)
    CHR$(231);: PEN 2: PRINT TAB(33) CHR$(231)
200 PEN 3: PRINT  TAB(10)  CHR$(231);:  PEN  2:  PRINT  TAB(33)
    CHR$(231)
210 PRINT TAB(12) CHR$(231);:PRINT TAB(15) "+";
220 PEN 1: PRINT  TAB(18)  CHR$(231);:  PEN  2:  PRINT  TAB(19)
    CHR$(231);
230 PEN 1: PRINT TAB(23) "=";  PRINT TAB(29) "+";:  PEN 1:  PRINT
    TAB(32) CHR$(231)
240 PEN 3: PRINT  TAB(10)  CHR$(231);:  PEN  2:  PRINT  TAB(33)
    CHR$(231)
250 PRINT TAB(12) CHR$(231);: PEN 3: PRINT TAB(26) CHR$(231);:
    PEN 2: PRINT TAB(33) CHR$(231)
260 PEN 1: PRINT  TAB(18)  CHR$(231);:  PEN  2:  PRINT  TAB(19)
    CHR$(231);
270 PEN 1: PRINT TAB(32) CHR$(231)
280 PEN 2: PRINT TAB(33) CHR$(231)
290 PRINT: PRINT
300 PEN 1: PRINT TAB(8) "FE O   +   3CO   =   2FE + CO"
310 PRINT TAB(10) "2 3                       2"
320 PRINT: PRINT "FINAL DE LA IMPRESION": END
```

5. PASO : Lista de variables

No se utiliza ninguna variable

6. PASO : Descripción del programa

Línea 10 - 60	Título
Línea 70	Visualización de bolita amarilla en columna 10
Línea 75	Visualización de bolita amarilla en columna 25 al lado una bolita azul
Línea 80	Visualización un signo + - en columna 12 seguido de 2 bolitas azules
Línea 85	Visualización un signo + - en la misma línea
Línea 86	Línea en blanco
Línea 90	Como línea 70
Línea 100	Como línea 75
Línea 110 - 130	Visualización de la ecuación de la reacción
Línea 140 - 150	Espera y borrado de la pantalla

Línea 160 - 280 Visualización gráfica de la segunda ecuación

Línea 290 - 310 Visualización de la ecuación correspondiente

Línea 320 Final del programa

7. PASO : Resultados

El programa presentado genera consecutivamente dos gráficos en la pantalla; el segundo dibujo aparece cuando el usuario pulsa CONT.

Podemos prescindir de presentar aquí ambos dibujos, pues no nos es posible efectuar una representación en colores. Nos parece más razonable que el lector pruebe directamente el programa en un televisor en color.

3.4 Cálculo estequiométrico

1. PASO : Presentación del problema

Los cálculos estequiométricos consisten en cuantificar las cantidades de las diferentes sustancias que participan en una reacción química. Permiten contestar por ejemplo a la siguiente cuestión:

¿Cuántos gramos de azufre son necesarios para convertir completamente en sulfuro de zinc 100 gramos de zinc? (Azufre = S; Zinc = Zn; Sulfuro de zinc = ZnS).

Peso atómico del azufre : 32.1
Peso atómico del zinc : 65.4

2. PASO : Análisis del problema

Para resolver el problema deberemos partir de la correspondiente reacción:

$$Zn + S \longrightarrow ZnS$$

Si relacionamos las llamadas masas molares de la substancia buscada y de las subtancias dadas, obtendremos que:

son necesarios 32.1 g de azufre para transformar 65.4 g de zinc (ver arriba).

Se cumple que:

$$x/100 = 32.1/65.4$$

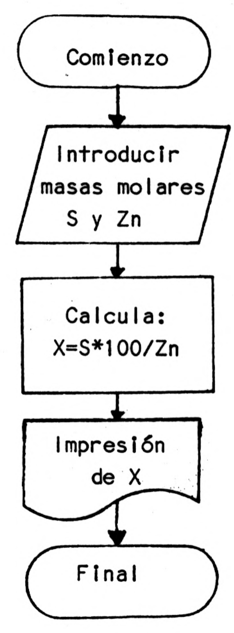
o

$$x = 3210/65.4$$

x es el valor buscado.

3. PASO : Diagrama de flujo

En este problema el diagrama de flujo vuelve a ser muy sencillo:



4. PASO : Programa

```
10 REM Q3-CALCULOS QUIM.
20 CLS
30 PRINT "PROGRAMA PARA CALCULOS ESTEQUIOMETRICOS"
40 PRINT: PRINT
50 PRINT: PRINT: PRINT TAB(9) "PROF. DR. W. VOSS, 1984": PRINT
60 PRINT: PRINT: PRINT "EJEMPLO :": PRINT: PRINT
70 PRINT "ZINC + AZUFRE = SULFURO DE ZINC :"
80 PRINT: PRINT " ZN  +    S    =    ZN S": PRINT: PRINT
90 PRINT: PRINT "CUANTOS GRAMOS DE AZUFRE SON NECESARIOS,"
100 PRINT "PARA TRANSFORMAR 100 GRAMOS DE ZINC COMPLETAMENTE "
110 PRINT "EN SULFURO DE ZINC?": PRINT: PRINT
120 INPUT "1. ELEMENTO : "; A$
130 INPUT "2. ELEMENTO : "; B$
140 PRINT: PRINT: PRINT
150 PRINT "PESO ATOMICO DE "; A$;: PRINT TAB(30) ":";:INPUT A
160 PRINT "PESO ATOMICO DE "; B$;: PRINT TAB(30) ":";:INPUT B
170 X = B * 100 / A
180 PRINT: PRINT: PRINT "CANTIDAD NECESARIA DE "; B$; ":": PRINT:
    PRINT TAB(5) X; " G"
190 PRINT: PRINT: PRINT "FINAL DEL CALCULO ": END
```

5. PASO : Lista de variables

A = Peso atómico elemento A
A$ = Nombre del elemento A

B = Peso atómico elemento B
B$ = Nombre del elemento B

X = Cantidad buscada

6. PASO : Descripción del programa

Línea 10 - 110 Título y visualización del ejemplo del que se ocupa el programa

Línea 120 - 130 Entrada de los nombres de los 2 elementos

Línea 140 3 Líneas en blanco

Línea 150 - 160 Visualización de los nombres de elementos y demanda de los pesos atómicos de los mismos

Línea 170 Cálculo de la cantidad buscada

Línea 180 Visualización del resultado (incluyendo líneas en blanco, que en un futuro no mencionaremos en todos los casos)

Línea 190 Final del programa

7. PASO : Resultados

Después de las aclaraciones, el programa pide el nombre del primer elemento.

Si introducimos p.e. "azufre", entonces nos pide el nombre del segundo elemento que deberá considerarse en el cálculo.

Si introducimos p.e. "zinc", el programa nos obliga a introducir los pesos atómicos del azufre y del zinc.

Si en respuesta introducimos los valores 32.1 y 65.4, el programa nos ofrece el siguiente resultado:

```
CANTIDAD NECESARIA DE ZINC:

   203.738317 G

FINAL DEL CALCULO
```

Se entiende que el programa debe ser modificado profundamente en determinados casos, cuando se trata de procesos químicos distintos a éste (p.e. también cuando participan más de dos substancias en las reacciones del problema).

En este caso hemos procurado únicamente presentar a modo de ejemplo un programa de este tipo.

3.5 El sistema periódico de los elementos

1. PASO : Presentación del problema

No sólo podemos utilizar un ordenador para efectuar cálculos como el del ejemplo precedente o para generar gráficos ilustrativos (como en el punto anterior), sino que puede utilizarse además- en el ámbito escolar - precisamente como "fichero de consulta".

El ejemplo del sistema periódico de los elementos ilustra el posible aspecto que cubre un ''programa de consulta'' de este tipo, que también puede entenderse como programa de entreno.

El usuario puede desear, por ejemplo, que el ordenador le dé informaciones básicas acerca de un elemento previamente introducido (p.e. el azufre).

Estas informaciones pueden ser por ejemplo:

1. Símbolo químico,
2. Número de orden en el sistema periódico,
3. Peso atómico,
4. Peso específico
5. Punto de fusión,
6. Punto de evaporación,
7. Grupo (I-VIII,0),
8. Subgrupo (a,b,),
9. Capas de electrones (K-Q)

2. PASO : Análisis del problema

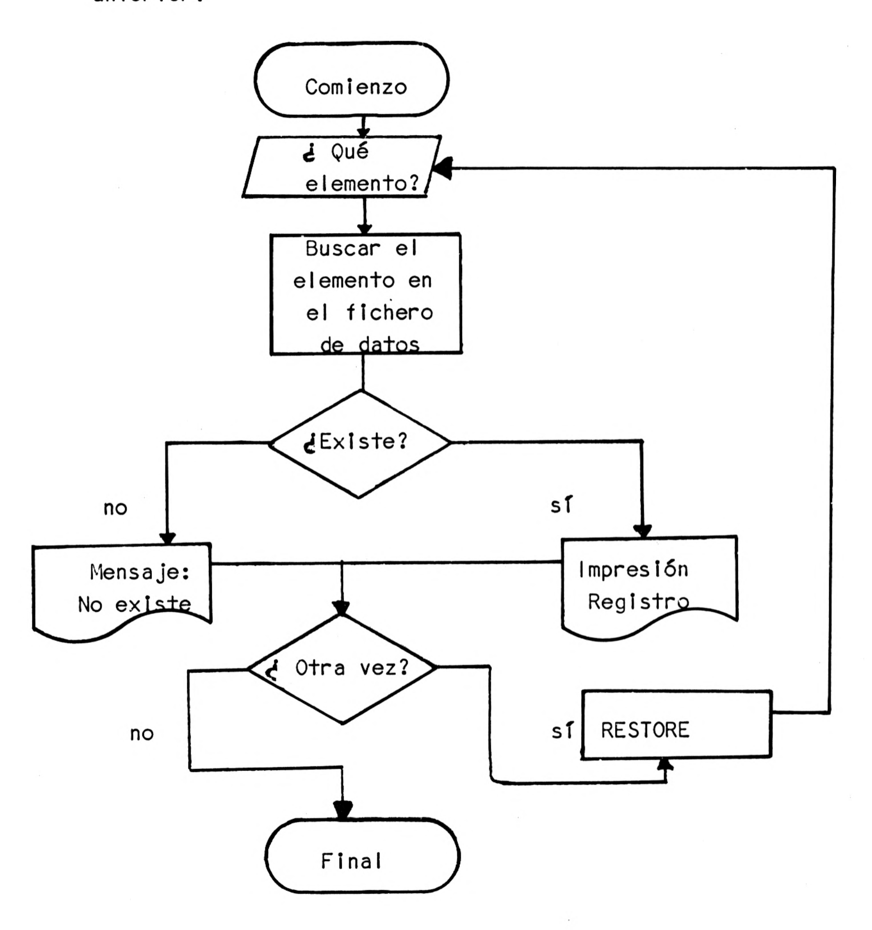
El desarrollo de un programa que nos pueda proporcionar estas informaciones no supone dificultades especiales.

El usuario indicará a través del INPUT el elemento del que desea obtener informaciones y el programa pondrá a su disposición el registro completo asignado a este elemento, extrayéndolo de los DATAS mediante la instrucción READ.

Finalmente puede consultar si se desean obtener las mismas informaciones de otro elemento químico. Si es así, puede accederse nuevamente al fichero mediante RESTORE.

3. PASO : Diagrama de flujo

El diagrama de flujo es igual de sencillo que el análisis del problema efectuado en el párrafo anterior:



4. PASO : Programa

```
10 REM Q4-ELEMENTOS
20 CLS
30 PRINT "PROGRAMA PARA INDICAR CARACTERISTICAS FUNDAMENTALES"
40 PRINT TAB(5) "DE LOS ELEMENTOS QUIMICOS"
50 PRINT: PRINT: PRINT TAB(9) "PROF. DR. W. VOSS, 1984": PRINT
60 N = 52: REM CANTIDAD DE ELEMENTOS REGISTRADOS
70 DIM N$(N,10)
75 FOR J = 1 TO 7
80 FOR I = 1 TO N
85 READ N$(I,J)
90 NEXT I
95 NEXT J
100 FOR I = 1 TO N
105 READ N$(I,8),N$(I,9),N$(I,10)
108 NEXT I
110 PRINT: INPUT "QUE ELEMENTO (NOMBRE) "; B$
120 FOR I = 1 TO N: IF B$ = N$(I,1) THEN 160
130 NEXT I
140 PRINT: PRINT "ESTE ELEMENTO NO SE ENCUENTRA "
150 PRINT "EN LA TABLA.": GOTO 290
160 CLS
170 PRINT N$(I,1);: PRINT TAB(23) N$(I,2)
180 PRINT: PRINT
190 PRINT "NUMERO DE ORDEN                  : ";N$(I,3)
200 PRINT "GRUPO                            : ";N$(I,8)
210 PRINT "SUBGRUPO                         : ";N$(I,9)
220 PRINT "CAPA                             : ";N$(I,10)
230 PRINT: PRINT
240 PRINT "PESOS                            : ";N$(I,4)
250 PRINT "ATOMICOS                         : ";N$(I,5)
260 PRINT
270 PRINT "PUNTO DE FUSION                  : ";N$(I,6)
280 PRINT "PUNTO EVAPORACION                : ";N$(I,7)
```

```
290 PRINT: PRINT: INPUT "OTRO ELEMENTO (S/N)";A$
300 IF A$ = "S" THEN CLS: GOTO 110
310 PRINT: PRINT "FINAL EN 310": END
495 REM DATOS Y ELEMENTOS
500 DATA ALUMINIO,ANTIMONIO,ARGON,ARSENICO,BARIO,PLOMO,BORO,
    BROMO,CLORO,CROMO
510 DATA HIERRO,FLUOR,ORO,HELIO,IRIDIO,IODO,CADMIO,POTASIO,CALCIO
    COBALTO
520 DATA CARBON,COBRE,LITIO,MAGNESIO,MANGANESO,MOLIBDENO,SODIO
530 DATA NEON,NIQUEL,FOSFORO,PLATINO,PLUTONIO,MERCURIO,RADIO,RADON
540 DATA OXIGENO,AZUFRE,SELENIO,PLATA,SILICIO,NITROGENO
550 DATA ESTRONCIO,TORIO,TITANIO,URANIO,VANADIO,HIDROGENO,BISMUTO,
    TUNGSTENO
570 DATA XENON,ZINC,ESTAÑO
595 REM DATOS ABREVIACIONES
600 DATA AL,SB,AR,AS,BA,PB,B,BR,CL,CR,FE,F,AU,HE,IR,I,CD,K,CA,CO
610 DATA C,CU,LI,MG,MN,MO,NA,NE,NI,P,PT,PU,HG,RA,RN,O,S,SE,AG,SI
620 DATA N,SR,TH,TI,U,V,H,BI,W,XE,ZN,SN
640 REM DATOS Y NUMEROS DE ORDEN
650 DATA 13,51,18,33,56,82,5,35,17,24,26,9,79,2,77,53,48,19,20,27
660 DATA 6,29,3,12,25,42,11,10,28,15,78,94,80,88,86,8,16,34,47,14
670 DATA 7,38,90,22,92,23,1,83,74,54,30,50
695 REM PESOS ATOMICOS
700 DATA 26.97,121.76,39.94,74.91,137.36,207.21,10.82,79.92,35.46
710 DATA 52.01,55.85,19,197,4.003,192.2,126.92,112.41,39.096,40.08
720 DATA 58.94,12.011,63.54,6.94,24.32,54.94,95.95,22.991,20.183
730 DATA 58.71,30.98,195.09,242,200.61,226.05,222,16,32.066,78.96
740 DATA 107.88,28.06,14.008,87.63,232.05,47.9,238.07,50.95,1.008
    209,183.86,131.3,65.38,118.7
745 REM DATOS CAPA
750 DATA 2.7,6.67,0.0018,5.72,3.5,11.34,1.73,3.14,1.557,7.14,7.86
760 DATA 0.0017,19.25,0.00018,22.42,4.942,8.64,0.862,1.545,8.83
770 DATA 3.51,8.933,0.534,1.75,7.3,10.2,0.971,0.0009,8.8,1.83
780 DATA 21.4,-,13.55,-,0.009,0.0014,2.07,4.46,10.5,2.34,0.00125
790 DATA 2.63,11.7,4.43,19,6.07,0.0009,9.8,19.3,0.0059,7.14,7.28
795 REM PUNTOS DE FUSION
```

```
800 DATA 658,630,-190,,704,327,2400,-7.3,-100,1800,1525,-218,1063
810 DATA -272,2454,114,321,64,851,1490,-,1083,180,650,1221,2622
820 DATA 99,-248,6,1452,44,1771,-,-39,700,-71,-218.7,113,220,960
830 DATA 1414,-210,771,1827,1727,1689,1726,-266,271,3380,-111.9,
    419,232
845 REM PUNTO DE EBULLICION
850 DATA 2270,1440,-186,616,1537,1690,2550,58,-34,2600,2450,-187
860 DATA 2677,-269,4400,184,770,762,1439,3185,-,2360,1372,1120
870 DATA 2152,4800,880,-246,3075,280,3800,-,357,1140,-61.9,-182.97
880 DATA 444,688,1930,2630,-195.8,1360,3530,-,-,3000,-272.78,1560,
    6000,-108.1,906,2430
900 REM CLASES
910 DATA III,B,M,V,B,O,Ø,A,M,V,B,N,II,A,P,IV,B,P,III,B,L,VII,B,N
920 DATA VII,B,M,VI,A,N,VIII,A,N,VII,B,L,I,B,P,Ø,A,K,VIII,A,P,VII
930 DATA B,O,II,B,O,I,A,N,II,A,N,VIII,A,N,IV,V,L,I,B,N,I,A,L,II,A
940 DATA M,VII,A,N,VI,A,O,I,A,M,Ø,A,L,VIII,A,N,V,B,M,VIII,A,P,-,-
950 DATA -,II,B,P,II,A,Q,Ø,A,P,VI,B,L,VI,B,M,VI,B,N,I,B,O,IV,B,M
960 DATA V,B,L,II,A,O,-,-,-,IV,A,N,-,-,-,V,A,N,I,A,K,V,B,P,VI,A,P,
    Ø,A,O,II,P,N,IV,B,O
```

5. PASO : Lista de variables

A$ = Cadena de respuesta (sí/no)
B$ = Campo de almacenamiento del elemento deseado
I = Indice
J = Indice
N = Cantidad de elementos
N$ = Elementos del fichero

6. PASO : Descripción del programa

Línea 10 - 50 Títulos

Línea 60 Indicación de la cantidad de elementos (en este programa no están incluídos todos los

	elementos sino únicamente 52, y para ampliar se debe modificar línea 60)
Línea 70	Dimensionado
Línea 75 - 108	Lectura de toda la información
Línea 110	Demanda del elemento del que se desea información
Línea 120 - 130	Búsqueda del elemento deseado; al encontrarlo se continúa en 160 de lo contrario en 140.
.ínea 140 - 150	Si el elemento no ha sido encontrado, se visualiza el mensaje correspondiente; bifurcando a línea 290.
.ínea 160	Borrado de pantalla
.ínea 170	Visualización del nombre del elemento y su abreviación.
ínea 180	Línea en blanco
ínea 190 - 280	Visualización de las informaciones restantes
ínea 290 - 300	Pregunta si se desea más información; en caso de 'SI', borrado de pantalla y bifurcación a 110

de lo contrario continúa en 310

Línea 310 Finalización del programa

Línea 495 - 960 Fichero en forma de 'DATAS'

7. PASO : Resultados

Introduzcamos por ejemplo el elemento Azufre a lo cual el ordenador contestará:

```
AZUFRE                   S

NUMERO DE ORDEN       : 16
GRUPO                 : VI
SUBGRUPO              : B
CAPA                  : M

PESO ATOMICO          : 32.066
PESO ESPECIFICO       : 2.07

PUNTO DE FUSION       : 112
PUNTO DE EBULLICION   : 444

              OTRO ELEMENTO (S/N)?
```

4

Capítulo 4 : Física

4.1 Consideración previa

El campo de la física a nivel escolar, ofrece problemas similares a los que ya conocemos del capítulo dedicado a las matemáticas (ver capítulo 2). En todo caso, aquí deben considerarse ciertas particularidades que nos obligan a hacer uso nuevamente de las posibilidades gráficas del ordenador.

Por esta razón - y ampliando lo dicho en el capítulo anterior con respecto a los métodos de programación de gráficos - en esta introducción profundizaremos otra vez en las instrucciones gráficas del BASIC.

4.2 Programación de gráficos

Si se utiliza el CPC 464 para la confección de gráficos, el centro de interés son los gráficos de alta resolución.

Según en que modalidad de pantalla trabaje el ordenador se dispone de una resolución de pantalla diferente (y como ya hemos mencionado, diferente gama de colores).

Con el comando siguiente se determina la modalidad de la pantalla:

Comando 8:

```
MODE número
```

En la posición de "número" se puede introducir 0, 1 ó 2.

Estos dígitos tienen el significado siguiente:

MODALIDAD	Caracteres por línea
0	20
1	40
2	80

Después de la puesta en marcha el ordenador se encuentra en modalidad 1 y se dispone de una pantalla de 40 caracteres por línea.

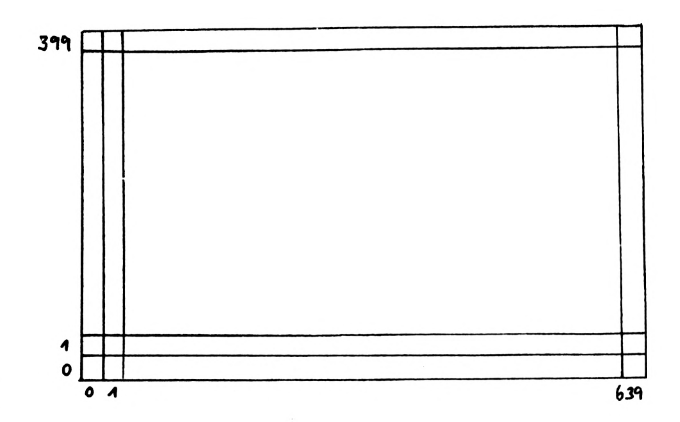
En general ofrece una alta resolución de gráficos con 600 * 400 puntos (640 columnas y 400 líneas).

Esta resolución nos permite producir líneas finas en la pantalla conociendo las correspondientes instrucciones.

Las instrucciones más importantes se explican a continuación y se comprueban con una serie de ejemplos simples.

Para confeccionar gráficos de alta resolución es aconsejable imaginarse que por la pantalla se está moviendo un cursor de gráficos (invisible), cuya posición indica en qué posición puede dibujarse en ese momento.

Ya hemos mencionado que hay que imaginarse la pantalla en modalidad gráficos según el dibujo siguiente.



Es muy importante marcar que la línea 0 es la línea inferior y la columna 0 se encuentra en el extremo izquierdo de la pantalla.

De ésta manera el punto de coordenadas (0,0), el origen del eje que puede imaginarse, se encuentra abajo a la izquerda, como acostumbramos dibujarlo en matemáticas.

Si se quiere dibujar un punto en la pantalla, sírvase de la instrucción siguiente:

Instrucción 17:

PLOT S,Z

La 'S' representa la columna (0-639) y 'Z' proporciona la línea (0-399).

El lector puede probar el siguiente programa:

```
10 CLS
20 PLOT 320,200
30 END
```

El resultado es un punto pequeño en el centro de la pantalla.

Igualmente fácil resulta tirar líneas rectas con la instrucción siguiente:

Instrucción 18:

DRAW S,Z

4.3 PROGRAMAS BASIC

La instrucción

DRAW 100,100

dibuja una línea recta desde la posición actual del cursor de gráficos hasta el punto de coordenadas (100,100).

Si el cursor de gráficos no se ha movido anteriormente mediante una instrucción PLOT o DRAW, esta línea se dibujará desde (0,0) hasta (100,100).

Supongamos que queremos dibujar una línea vertical en el centro de la pantalla. Para ello necesitamos el programa siguiente:

```
10 CLS
20 PLOT 320,0
30 DRAW 320,399
40 END
```

De la misma manera se procede en un programa para dibujar una línea vertical en el centro de la pantalla:

```
10 CLS
20 PLOT 0,200
30 DRAW 639,200
40 END
```

Igual de fácil es dibujar un par de ejes coordenados:

```
10 CLS
20 PLOT 0,0
30 DRAW 400,0
40 PLOT 0,0
50 DRAW 0,250
60 END
```

Con este ordenador a diferencia de muchos otros es muy fácil introducir textos en una representación gráfica.

Los gráficos pueden confeccionarse también utilizando caracteres especiales que ofrece el código ASCII, para el CPC 464 /6128

Estos caracteres se encuentran en el área de los códigos 123 a 255, y pueden llamarse mediante la instrucción siguiente:

```
PRINT CHR$(Número)
                        Para número = 123 a 255
```

El lector puede probar el programa siguiente, que visualiza todos los caracteres especiales en la pantalla. El programa se detiene después de visualizar 15 carqacteres, para que el usuario pueda estudiar los símbolos. Para continuar se debe entrar el comando

```
CONT
```

que permite visualizar entonces los 15 caracteres siguientes.

```
10 CLS
20 PRINT "NRO.   SIMBOLO":PRINT
30 PRINT
40 FOR I = 123 TO 255
50 PRINT I; "      "; CHR$(I)
60 IF I / 15 = INT(I/15) THEN STOP
70 NEXT I
80 END
```

También se puede 'dibujar' con los caracteres especiales, que se proporcionan con la función CHR$.

Por ejemplo bajo el código 124, encontramos una raya vertical de aproximadamente 5 milímetros de longitud.
Si programamos ahora:

```
10 CLS

20 PRINT: PRINT: PRINT

30 FOR I = 1 TO 10

40 PRINT TAB(10) CHR$(124)

50 NEXT I

60 END
```

Obtenemos una raya vertical sobre 10 líneas en la columna 10 de la pantalla, encontrandose la misma con modalidad de gráficos normal, es decir 40 columnas y 25 líneas.

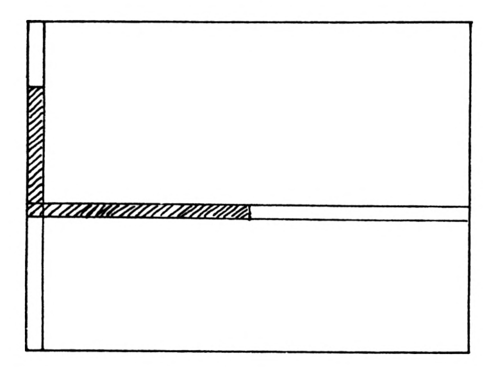
Se puede proceder de igual forma con otros símbolos utilizando la función CHR$.

Dibujando con estos símbolos la modalidad 1 nos proporciona en la pantalla 40 columnas y 25 líneas.

En este caso no se trata de gráficos de alta resolución como podemos ver en el ejemplo siguiente:

```
10 CLS
20 PRINT: PRINT: PRINT: PRINT
30 FOR I = 1 TO 10
40 PRINT CHR$(143)
50 NEXT I
60 FOR I = 1 TO 20
70 PRINT CHR$(143);
80 NEXT I
90 PRINT: PRINT: PRINT "FIN": END
```

Este programa produce el siguiente dibujo:



Naturalmente pueden alterarse los colores con esta modalidad de gráficos.
Si por ejemplo introducimos al programa anterior las líneas:

```
15 PEN 2
```

y

```
85 PEN 1
```

el eje de coordenadas producido se visualizará en azul cielo y la expresión final "FIN" nuevamente en amarillo.

Comprobando el programa anterior el lector notará que el desplazamiento de la coordenada se ha logrado mediante una serie de instrucciones 'PRINT', para producir líneas en blanco.

Estos desplazamientos o posicionamientos se pueden lograr con mayor facilidad con la instrucción siguiente:

Instrucción 19:

```
nn LOCATE  S,Z
```

Esta instrucción mueve el cursor a la posición de la pantalla indicada por columna 'S' (1-40) y línea 'Z' (1-25).

Si queremos por ejemplo escribir la palabra 'OLE' en el centro de la pantalla debe utilizarse el programa siguiente:

```
10 CLS
20 LOCATE 19,12
30 PRINT "OLE"
40 END
```

Esta instrucción LOCATE, nos permite de manera fácil mezclar gráficos con textos, lo que en otros ordenadores -por lo menos en gráficos de alta resolución- puede resultar laborioso.

Por ejemplo se quieren denominar los ejes en un sistema de coordenadas, con los vocablos 'ENTRADAS' (eje horizontal) y 'CANTIDAD' (eje vertical).

Esto puede realizarse con el programa siguiente:

```
10 CLS
20 LOCATE 1,10
30 PRINT "CANTIDAD"
40 LOCATE 25,22
50 PRINT "ENTRADAS"
60 PLOT 50,50
70 DRAW 350,50
80 PLOT 50,50
90 DRAW 50,220
100 END
```

Se entiende que según esta muestra pueden asimismo resolverse otros ejercicios.

Después de esta introducción para la programación de gráficos, nos dirigimos a problemas concretos.

4.4 El dinamómetro

1. PASO : Presentación del problema

Si colgamos diferentes objetos de un dinamómetro, éste se estira en mayor o menor grado, según sea el peso de los cuerpos.

Así se cumple una ley de proporcionalidad que dice: si doblamos el peso, el alargamiento del dinamómetro se duplica.

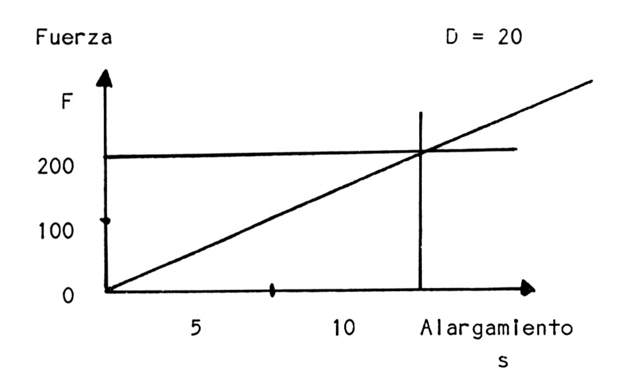
Existe también una llamada constante de muelle, que es igual ai cociente entre el peso F y el alargamiento del muelle s, es decir:

$$D = F/s = \text{constante}$$

Si, por ejemplo, D = 20, una fuerza de 100 (cN) producirá un alargamiento de 5 cm, y una fuerza de 200 (cN) un alargamiento del muelle de 10 cm etc.

2. PASO : Análisis del problema

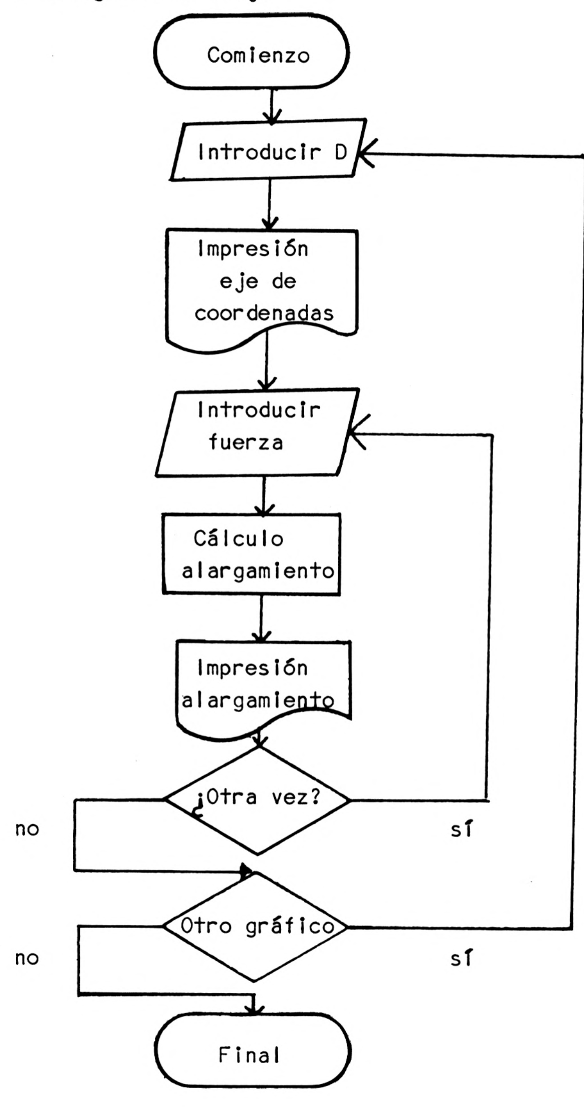
La relación antes descrita puede ser representada gráficamente (p.e. para D = 20):



El programa BASIC a desarrollar, debe imprimir el gráfico correspondiente a diversos valores de la constante de muelle D en la pantalla, siendo necesario estructurarlo del modo siguiente:

- Trazado de unos ejes de coordenadas,
- introducción de un valor arbitrario para D,
- impresión de una recta de constancia (ver arriba),
- contestación a la pregunta referente al alargamiento del dinamómetro tras introducir un peso cualquiera.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM F1-DINAMOMETRO
20 CLS
30 PRINT "PROGRAMA PARA ILUSTRAR GRAFICAMENTE"
40 PRINT TAB(3) "LA LEY DE HOOKE"
50 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W.VOSS, 1984":PRINT
55 PRINT: PRINT: PRINT: PRINT
60 INPUT "VALOR DE LA CONSTANTE DE MUELLE :"; D
70 CLS: REM SISTEMA DE COORDENADAS
75 LOCATE 3,3: PRINT "FUERZA"
77 LOCATE 20,3: PRINT " D = :;D
80 PLOT 50,50: DRAW 380,50
90 PLOT 50,50: DRAW 50,350
100 LOCATE 25,22: PRINT "CM"
110 PLOT 50,50
120 B = D / 10
125 A=380
130 E=50+B*350
140 IF E>399 THEN A = 400 - (350 * (E - 399) / (E - 49)): E = 399
150 DRAW A,E
263 FOR I = 1 TO 9999: NEXT I
265 CLS
270 INPUT "FUERZA  F                         : ";F
280 FW = F / D
290 PRINT: PRINT: PRINT "CONSTANTE DE MUELLE                : "; D
300 PRINT: PRINT: PRINT "RECORRIDO                          : "; FW
310 PRINT: PRINT: PRINT: INPUT "OTRO CALCULO (S/N) "; A$
320 IF A$ = "S" THEN 265
330 PRINT: PRINT: INPUT "OTRO GRAFICO (S/N) "; A$
340 IF A$ = "S" THEN CLS: GOTO 60
350 PRINT: PRINT: PRINT "FINAL": END
```

5. PASO : Lista de variables

A = Columna para el trazado de la recta

A$ = Cadena de respuesta (sí/no)

B = D/10 (Magnitud para normalizar la impresión en pantalla)

D = Constante de muelle

E = Línea para el trazado de la línea de constancia

F = Fuerza

FW = Recorrido del muelle

I = Indice variable

6. PASO : Descripción del programa

Línea 10 - 55	Título
Línea 60	Entrada constante de muelle D
Línea 70 - 100	Trazado de los ejes de coordenadas. Origen: 50,50 Eje horizontal: 50,50 hasta 380,50 Eje vertical: 50,50 hasta 50,350 Denominación de los ejes dibujados (línea 75 y línea 100) y denominación de la función
Línea 110 - 150	Trazado de la recta desde 50,50 hasta A,E Deben alterarse los valores de origen para A (A=380; línea 125) y para E, si E fuera mayor que 399.
Línea 263 - 265	Bucle de espera y borrado de la pantalla
Línea 270	Demanda de la fuerza F
Línea 280	Cálculo del recorrido del muelle

Línea 290 - 300	Visualización de resultados
Línea 310	Consulta si se desea un nuevo cálculo; en caso afirmativo, retorna a 265 en caso negativo, continúa en 330
Línea 330	Consulta si se desea otro gráfico en caso afirmativo, retorna a línea 60 tras borrar la pantalla en caso negativo, seguir en línea 350
Línea 350	Final del programa

7. PASO : Resultados

Después de introducir la información mediante INPUT, el programa genera un gráfico tal como el que ya presentamos esquemáticamente en una de las páginas precedentes.

Después de una pausa, el programa pide la fuerza F y calcula el correspondiente recorrido de muelle. Sobra aquí cualquier ejemplo de cálculo (comparar con el 1. PASO, que contiene ya un ejemplo).

4.5 El principio de Arquímedes

1. PASO : Presentación del problema

El principio de Arquímedes dice lo siguiente:

El empuje que recibe un cuerpo sumergido en un fluído es igual al peso del líquido que desplaza.

Este principio describe el hecho conocido de que la fuerza necesaria para apretar hacia abajo un cubo vacío dentro de un líquido aumenta progresivamente.

A medida que un cuerpo se sumerge en un líquido, éste se ve sometido a una pérdida de peso aparente cada vez mayor, debida al creciente empuje.

Presentamos a continuación un programa que, basándose en este principio, calcula el volumen y la densidad (peso específico) de un cuerpo desconocido sumergido en un líquido conocido (p.e. agua), una vez se conoce su peso en el aire.

2. PASO : Análisis del problema

Como en todos los demás problemas, aquí se procederá de acuerdo con el probado esquema:

E = Entrada

P = Proceso

S = Salida

Es decir, en primer lugar estudiaremos qué informaciones de entrada necesita el ordenador si debe resolver el problema antes formulado. Necesita lo siguiente:

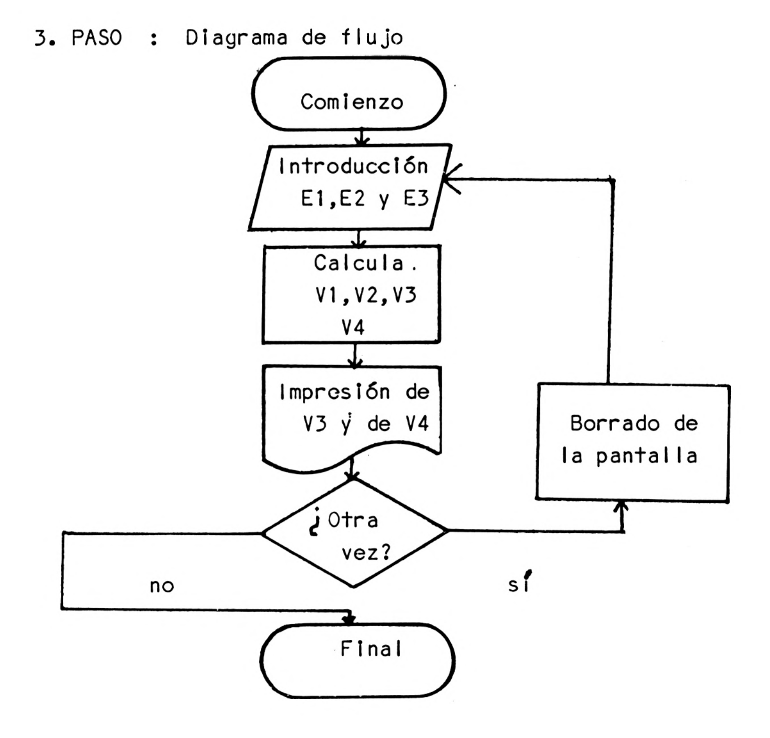
E1. El peso del cuerpo desconocido en el aire;
E2. el peso aparente del cuerpo dentro del fluído, en el que se sumerge;
E3. el peso específico del líquido utilizado para la comprobación.

Cuando ya disponemos de estas informaciones, puede determinarse mediante el proceso lo siguiente:

V1. El empuje y el peso de la cantidad de líquido desplazada;
V2. el volumen de esta cantidad de líquido;
V3. el volumen del cuerpo desplazante de prueba;
V4. El peso específico del cuerpo de prueba sumergido.

Finalmente deberán imprimirse como resultados los obtenidos en los últimos puntos V3 y V4.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM F2-ARQUIMEDES
20 CLS
30 PRINT "PRINCIPIO DE ARQUIMEDES"
40 PRINT: PRINT
50 PRINT TAB(9) :PROF.DR.W.VOSS, 1984":PRINT
55 PRINT: PRINT "CON EL PRINCIPIO DE ARQUIMEDES"
56 PRINT "PUEDEN DETERMINARSE EL VOLUMEN Y LA"
57 PRINT "DENSIDAD DE UN CUERPO DESCONOCIDO"
75 PRINT "INTRODUZCA LAS SIGUIENTES INFORMACIONES  : ": PRINT
90 PRINT TAB(5) "PESO DEL CUERPO EN EL AIRE"
100 PRINT TAB(5) "   (PA)"
110 PRINT: PRINT TAB(5) "PESO APARENTE DEL CUERPO"
120 PRINT TAB(5) "EN EL FLUIDO (PF)  "
130 PRINT: PRINT TAB(5) "DENSIDAD DEL FLUIDO (DF) "
140 PRINT: PRINT: INPUT "PA = "; PA
150 PRINT: INPUT "PF = "; PF
160 PRINT: INPUT "DF = ";DF
170 PP = PA - PF
175 VD = PP / DF
180 VC = VD
190 DC = PA / VC
210 PRINT "CUERPO DE ESTUDIO   : ": PRINT
220 PRINT: PRINT "    VOLUMEN          : "; VC
230 PRINT: PRINT "    DENSIDAD         : "; DC
240 PRINT: PRINT: PRINT: INPUT "OTRA VEZ (S/N) "; A$
250 IF A$ = "S" THEN CLS: GOTO 75
260 PRINT: PRINT "FINAL DEL CALCULO": END
```

5. PASO : Lista de variables

A$ = Variable de cadena para las respuestas sí/no

PA = Peso del cuerpo en el aire
PF = Peso aparente del cuerpo en el fluído
PP = Pérdida aparente de peso

VC = Volumen del cuerpo
VD = Volumen del líquido desplazado

DF = Peso específico del líquido
DC = Peso específico del cuerpo

6. PASO : Descripción del programa

Línea 10-130 : Título y anotaciones aclaratorias; dimensionado de A$

Línea 140-160 : Demanda de las informaciones INPUT

Línea 170-190 : Pasos del proceso (cálculos)

Línea 200-230 : Impresión de los resultados

Línea 240-260 : Consulta si ha de seguir calculándose (vuelta a 75) o final del programa

7. PASO : Resultados

Si introducimos por ejemplo las siguientes informaciones INPUT:

E1: Peso del cuerpo = 70 g
E2: Peso aparente del cuerpo en el fluído de prueba (a medir con un dinamómetro) = 40 g;
E3: Peso específico del fluido de prueba = 1 (agua),

el programa genera estos resultados:

```
CUERPO DE ESTUDIO :

     VOLUMEN     = 30

     PESO ESP.   = 2.3333333

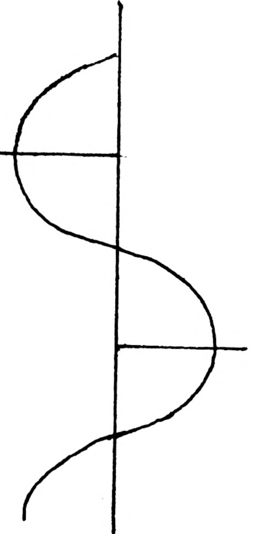
OTRA VEZ (S/N) ?
```

4.6 Movimiento pendular

1. PASO : Presentación del problema

El movimiento pendular es un fenómeno físico básico que juega un papel importante en la acústica, la óptica y en la electricidad.

Los movimientos pendulares siguen una oscilación sinusoidal, como se muestra en el esquema contiguo.



Presentamos ahora un programa, que representa gráficamente una oscilación sinusoidal para diversas amplitudes (A) y periodos (P), que a su vez determinan la llamada frecuencia (F) de cualquier oscilación.

2. PASO : Análisis del problema

Para el programa previsto deben introducirse la amplitud y el periodo de la oscilación como informaciones INPUT.

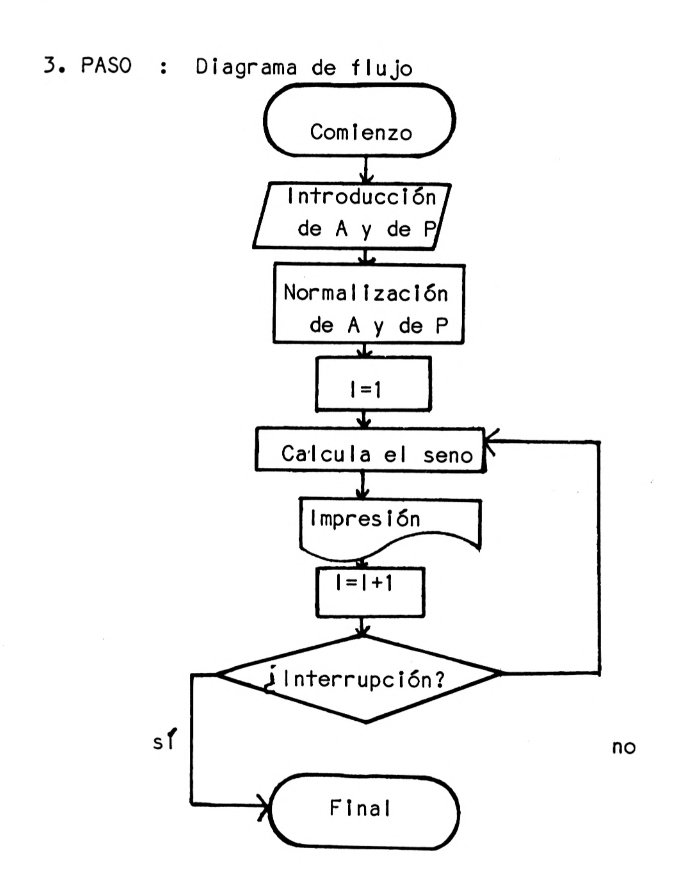
El programa puede ocuparse entonces de que la oscilación sinusoidal "pase", recorriendo la pantalla de arriba a abajo como en el esquema anterior (cosa que resulta más sencilla de programar que si imprimimos la oscilación horizontalmente).

Fundamentalmente, el contenido del programa debe centrarse en la normalización de la amplitud y del período para aprovechar óptimamente la representación gráfica de la oscilación sobre la pantalla.

Esto se consigue colocando la línea central de la oscilación en la columna 20 y ajustando la amplitud, es decir la mayor separación de esta línea, p.e. al valor máximo de 18.

Como puede deducirse de la distribución que acabamos de realizar, no seleccionamos los gráficos de alta resolución (ver Cap. 9), sino que representamos la oscilación con ayuda de PRINT CHR$(20).

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM F3-PENDULO
20 CLS
30 PRINT "PROGRAMA PARA LA SIMULACION"
40 PRINT "DE UN MOVIMIENTO PENDULAR""PRINT: PRINT
50 PRINT: PRINT TAB(9) "PROF.DR.W.VOSS, 1984": PRINT
52 PRINT: PRINT "PARA INTERRUMPIR PULSAR TECLA ESC"
54 PRINT: PRINT "PARA FINALIZAR DEFINITIVAMENTE PULSAR TECLA"
56 PRINT TAB(10) "ESC- 2 VECES"
60 INPUT "AMPLITUD DE LA OSCILACION  : "; A
70 PRINT: INPUT "PERIODO                 : "; P
75 IF A>19 THEN A = 19
80 CLS
90 I = 1
100 J = (2 * (22 / 7) / P) * I
110 Y = SIN(J)
120 S = 20 + Y * A
130 FOR K = 1 TO S
140 PRINT CHR$(231);
150 NEXT K
160 PRINT
170 I = I + 1: GOTO 100
```

5. PASO : Lista de variables

A = Amplitud de la oscilación

I = Indice variable

J = Valor transformado: J = (2*pi/p)*I
(J sirve para normalizar el período)

K = Indice variable que fija la longitud de las cadenas de puntos visualizados por línea de pantalla con ayuda del valor S.

P = Período de la oscilación.
S = Posición de memoria de la pantalla, en la que finaliza cada línea
Y = SEN(J)

6. PASO : Descripción del programa

Línea 10 - 55	Título y aclaraciones
Línea 60 - 70	Demanda de informaciones INPUT
Línea 75	Limitación de la amplitud de la oscilación al valor máximo de 18.
Línea 80	Borrado de la pantalla
Línea 90	Comienzo del bucle de impresión (I = índice de conteo de líneas)
Línea 100	Normalización de I después de tomar el período P con la magnitud auxiliar J
Línea 110	Determinación del correspondiente valor seno
Línea 120	Determinación de la posición de línea de cada valor de la función en la pantalla, teniendo en cuenta que el valor cero debe estar en la columna 20
Línea 130 - 150	Bucle para rellenar la línea con "bolitas" hasta el valor correspondiente de la función
Línea 160 - 170	Se fuerza un cambio de línea y se imprime la siguiente línea saltando nuevamente a 100 e incrementando en 1.

7. PASO : Resultados

Los resultados que ofrece este programa son análogos al croquis del 1. PASO, con la diferencia de que el espacio entre el margen izquierdo de la pantalla y la sinusoide está lleno de "bolitas" claras.

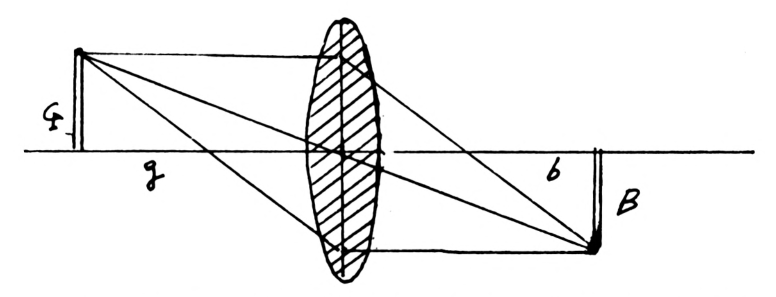
El lector puede descubrir por sí mismo cómo hay que modificar el programa para suprimir este "rellenado", es decir, cómo hay que hacer para marcar ópticamente con "bolitas" sólo aquellas posiciones que realmente se encuentran sobre la sinusoide.

El programa aquí presentado es un programa sinfín, que sólo puede interrumpirse pulsando la tecla ESC. Esta característica también puede modificarse fácilmente.

4.7 Reproducción óptica

1. PASO : Presentación del problema

Una lente convexa nos permite obtener la imagen de cualquier objeto - tal como se muestra en la ilustración siguiente:



La cámara fotográfica funciona según este principio, así como también el ojo humano.

Presentamos a continuación un programa que calcula el tamaño de la imagen y la escala de reproducción de una lente convexa para un tamaño del objeto, una distancia del mismo y una distancia focal de la lente dados.

2. PASO : Análisis del problema

El programa a realizar necesita las siguientes informaciones INPUT:

1. Tamaño del objeto,
2. distancia del objeto,
3. distancia focal de la lente.

Para resolver el problema necesitamos la llamada ecuación óptica:

$$1/f = 1/g + 1/b$$

donde:
- f = distancia focal
- g = distancia del objeto
- b = distancia de la imagen

Despejando b en esta ecuación se obtiene:

$$b = \frac{1}{1/f-1/g}$$

Se cumple también que la escala de reproducción A es igual a:

$$A = B/G = b/g$$

Si el objeto a reproducir tiene por ejemplo un tamaño G = 20 cm, puede determinarse también el tamaño de la imagen con ayuda de las igualdades citadas:

$$B = (b*G)/g, \text{ porque } B/G = b/g$$

3. PASO : Diagrama de flujo

4. PASO : Programa

```
10 REM F4-LENTE
20 CLS
30 PRINT "PROGRAMA PARA CALCULAR LA DISTANCIA,"
40 PRINT "IMAGEN, TAMANO DE LA IMAGEN Y"
50 PRINT "ESCALA DE REPRODUCCION DE UNA"
60 PRINT "LENTE CONVEXA": PRINT
65 PRINT: PRINT TAB(9) "PROF.DR.W.VOSS, 1984": PRINT
70 PRINT: PRINT "INTRODUZCA LOS DATOS SIGUIENTES:": PRINT
80 PRINT TAB(5)  " - TAMANO DEL OBJETO    : ";: INPUT G
90 PRINT TAB(5)  " - DISTANCIA DEL OBJETO : ";: INPUT GW
100 PRINT TAB(5) " - DISTANCIA FOCAL      : ";: INPUT F
110 CLS
120 BW = 1 / (1/F - 1/GW)
130 A = BW / GW
140 B = (G*BW) / GW
145 BW = INT(BW*100+0.5)/100: A = INT(A*100+0.5)/100:
        B=INT(B*100+0.5)/100
150 PRINT "DATOS DE PARTIDA : ": PRINT
160 PRINT TAB(5)        "DISTANCIA FOCAL          : "; F
170 PRINT: PRINT TAB(5) "TAMANO DEL OBJETO        : "; G
180 PRINT: PRINT TAB(5) "DISTANCIA DEL OBJETO     : "; GW
190 PRINT: PRINT: PRINT "RESULTADOS     : ": PRINT
200 PRINT: PRINT TAB(5) "ESCALA DE REPRODUCCION   : "; A
210 PRINT: PRINT TAB(5) "TAMANO DE LA IMAGEN      : "; B
220 PRINT: PRINT TAB(5) "DISTANCIA DE LA IMAGEN   : "; BW
230 PRINT: PRINT: PRINT: INPUT "OTRO CALCULO (S/N) "; A$
240 IF A$ = "S" THEN CLS: GOTO 70
250 PRINT: PRINT "FINAL DE LOS CALCULOS": END
```

5. PASO : Lista de variables

A = Escala de reproducción

A$ = Variable de cadena (sí/no)
B = Tamaño de la imagen
BW = Distancia de la imagen
F = Distancia focal
G = Tamaño del objeto
GW = Distancia del objeto

6. PASO : Descripción del programa

Línea 10 - 70	Título y aclaraciones
Línea 80 - 100	Demanda de informaciones INPUT
Línea 110 - 140	Borrado de pantalla y cálculos
Línea 145	Redondeo a 2 posiciones decimales
Línea 150 - 220	Impresión de los resultados
Línea 230 - 240	Consulta si se desea calcular más (en cuyo caso retorna a línea 70 después de borrar la pantalla)
Línea 250	Finalización del programa

7. PASO : Resultados

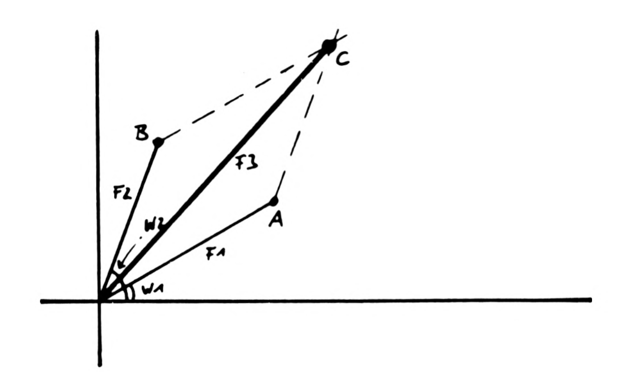
Si introducimos por ejemplo: G = 20 cm, GW = 300 cm, F = 15 cm, obtendremos:

```
        DISTANCIA FOCAL          : 15
        TAMAÑO DEL OBJETO        : 20
        DISTANCIA DEL OBJETO     : 300
    RESULTADOS:
        ESCALA DE REPRODUCCION   : .05
        TAMAÑO DE LA IMAGEN      : 1.05
        DISTANCIA DE LA IMAGEN   : 15.79
    MAS CALCULOS (S/N)
```

4.8 Paralelogramo de fuerzas

1. PASO : Presentación del problema

Cuando se aplican dos fuerzas sobre un punto común, se obtiene una fuerza reultante, que es la diagonal del llamado paralelogramo de fuerzas, como muestra el esquema siguiente:



En el programa que sigue se efectúa la representación gráfica del paralelogramo de fuerzas para cualquier constelación posible de dos fuerzas y se calcula la fuerza resultante F3.

2. PASO : Análisis del problema

El punto de aplicación de ambas fuerzas debe emplazarse en el centro de la pantalla (160,80).

Se introducen las fuerzas F1 y F2, que en primer lugar se convertirán de modo que F1 + F2 no sea mayor que 320, para que el espacio de la pantalla resulte suficiente incluso con una constelación poco favorable.

Deben introducirse además los ángulos W1 y W2 con los que se aplican las fuerzas F1 y F2 en el punto de aplicación.

Con ayuda del teorema de Pitágoras y de las relaciones entre ángulos del triángulo rectángulo pueden calcularse la columna y la línea de A y B, y a partir de esto también las coordenadas de C (ver gráfico).

Si se conoce la posición de C puede calcularse asimismo la fuerza F3 resultante de F1 y F2 (teorema del coseno).

3. PASO : Diagrama de flujo

Aquí puede prescindirse del diagrama de flujo.

4. PASO : Programa

```
10 REM F5-PALALELOGRAMO
20 CLS
30 PRINT "PROGRAMA PARA DIBUJAR UN PARALELOGRAMO"
40 PRINT "DE FUERZAS PARA 2 FUERZAS APLICADAS"
50 PRINT: PRINT
60 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W.VOSS, 1984": PRINT
70 SU = 320: ZU = 200
80 PRINT: PRINT: PRINT: PRINT
90  INPUT "PRIMERA FUERZA F1                  : "; F1
100 INPUT "ANGULO DE APLICACION F1 (0-360)    : "; W1
105 PRINT: PRINT
110 INPUT "SEGUNDA FUERZA  F2                 : "; F2
120 INPUT "ANGULO DE APLICACION F2 (0-360)    : "; W2
130 W1 = W1 * ((22/7)/180): W2 = W2 * ((22/7)/180)
135 H = F1 : G = F2
140 F1 = (F1/(F1+F2))*300: F2 = (F2/(H+F2))*300
150 Z1 = F1 * SIN(W1) + 200
160 S1 = ((Z1-200) / TAN(W1)) + 320
165 CLS
170 PLOT 320,200
180 DRAW S1,Z1
190 Z2 = F2 * SIN(W2) + 200
200 S2 = ((Z2-200) / TAN(W2)) + 320
210 PLOT 320,200
220 DRAW S2,Z2
230 S3 = S2 + (S1-SU)
240 Z3 = Z2 + (Z1 - ZU)
250 PLOT SU,ZU
260 DRAW S3,Z3
270 F3 = SQR((Z3-ZU)^2 + (S3-SU)^2)
275 F3 = F3 * (G+H) / 300
280 LOCATE 5,16
```

```
290 PRINT "FUERZA 1  =  "; H
300 PRINT TAB(5) "FUERZA 2   =  "; G
310 PRINT: PRINT "FUERZA RESULTANTE  = "; F3
320 PRINT: PRINT: PRINT "OTRO CALCULO (S/N) "; A$
330 IF A$ = "S" THEN CLS: GOTO 90
340 PRINT: PRINT: PRINT "FINAL": END
```

5. PASO : Lista de variables

A$	Variable de string (Sí/No)
F1	Primera fuerza
F2	Segunda fuerza
F3	Tercera fuerza
G	Magnitud auxiliar (=F2)
H	Magnitud auxiliar (=F1)
SU	Columna del origen
S1	Columna punto A
S2	Columna punto B
S3	Columna punto C
W1	Primer ángulo de aplicación
W2	Segundo ángulo de aplicación
ZU	Línea del origen
Z1	Línea punto A
Z2	Línea punto B
Z3	Línea punto C

6. PASO : Descripción del programa

Línea 10 - 60	Títulos
Línea 70	Coordenadas de origen

Línea 80 - 120	Introducción de las informaciones de partida
Línea 130	Cnversión del ángulo en unidades de parámetro de círculo
Línea 135	Almacenamiento de los valores de fuerza introducidos
Línea 140	Normalización de ambas fuerzas para el mejor aprovechamiento de la pantalla
Línea 150 - 160	Determinación de coordenadas de A
Línea 165 - 180	Trazado de F1
Línea 190 - 200	Coordenadas de B
Línea 210 - 220	Trazado de F2
Línea 230 - 240	Coordenadas de C
Línea 250 - 260	Trazado de F3
Línea 270	Cálculo de F3 (Pitágoras)
Línea 275	Normalización de F3 al igual que línea 140
Línea 280 - 310	Visualización de los resultados
Línea 320 - 340	Final del programa con la alternativa para continuar con otro cálculo (retornando a línea 90)

7. PASO : Resultados

Prescindimos aquí de ofrecer resultados, dado que el esquema del 1. PASO es suficientemente claro.

4.9 La Ley de OHM

1. PASO : Presentación del problema

La ley de Ohm se cumple en aquellos conductores eléctricos en los que la corriente crece proporcionalmente a la diferencia de potencial.

Esta ley dice lo siguiente:

$$\frac{\text{diferencia de potencial}}{\text{intensidad}} = \text{constante} = R$$

La magnitud R se denomina resistencia eléctrica del conductor.

En el programa siguiente se calcula para cada par de valores (dif. potencial, intensidad, resistencia), la tercera magnitud.

En esta visualización se introduce la instrucción 'PRINT USING'.
La línea:

```
X = 12.323:PRINT USING "  .  "; X
```

presenta una salida 12.32.
La máscara "   " (u otra) los resultados se visualizan de diferente manera.

2. PASO : Análisis del problema

El programa que nos ocupa no ofrece ninguna dificultad desde el punto de vista matemático.

La única dificultad estriba en la forma lógica de programar, permitiendo la entrada de una de las 3 variables diferentes, y la combinación de las mismas.

Existen 3 posibilidades de combinación:

1. Diferencia de potencial e intensidad
2. Diferencia de potencial y resistencia eléctrica
3. Intensidad y resistencia eléctrica.

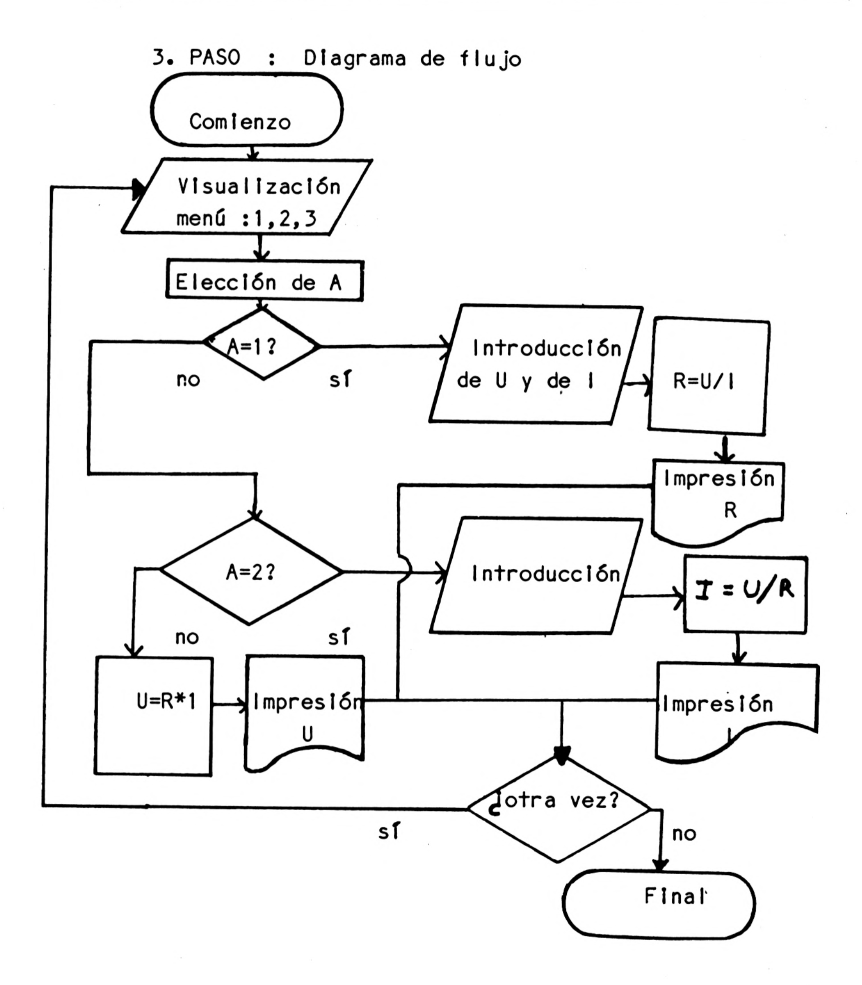
Para resolver este problema, se confecciona un menú técnico.

Mediante este menú, que se visualiza en primer término a través de la pantalla, se definen las variantes que determinan el camino lógico.

Este menú tendrá en nuestro caso 3 posiciones.

En el siguiente diagrama de flujo se puede observar el camino para resolver nuestro problema.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM F6-LEY DE OHM
20 CLS
30 PRINT "PROGRAMA PARA MOSTRAR LAS RELACIONES ENTRE"
40 PRINT "DIF. DE POTENCIAL, INTENSIDAD Y RESISTENCIA"
50 PRINT TAB(7) ELECTRICA (LEY DE OHM)."
60 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W.VOSS, 1984": PRINT
70 PRINT: PRINT "CON QUE DATOS SE TRABAJA? ": PRINT
80 PRINT          TAB(5) "DIFERENCIA POTENCIAL E INTENSIDAD   (1)"
90 PRINT: PRINT  TAB(5) "DIFERENCIA POTENCIAL Y RESISTENCIA  (2)"
100 PRINT: PRINT TAB(5) "INTENSIDAD Y RESISTENCIA            (3)"
110 PRINT: PRINT: INPUT "INTRODUZCA UNA CIFRA POR FAVOR "; A
120 CLS
130 IF A > 1 THEN 190
140 INPUT "DIFERENCIA POTENCIAL       : "; U
150 INPUT "INTENSIDAD                 : "; I
160 R = U / I
170 PRINT: PRINT: PRINT "RESISTENCIA     : "; USING "#####.##"; R
180 GOTO 300
190 IF A = 3 THEN 250
200 INPUT "DIFERENCIA POTENCIAL       : "; U
210 INPUT "RESISTENCIA                : "; R
220 I = U / R
230 PRINT: PRINT: PRINT "INTENSIDAD    : "; USING "#####.##"; I
240 GOTO 300
250 INPUT "RESISTENCIA                : "; R
260 INPUT "INTENSIDAD                 : "; I
270 U = R*I
280 PRINT: PRINT: PRINT "DIF.  POT.    : "; USING "#####.##"; U
300 PRINT: PRINT: PRINT: INPUT "MAS CALCULOS (S/N)? "; A$
310 IF A$ = "S" THEN CLS: GOTO 70
320 PRINT: PRINT "FINAL DEL CALCULO":END
```

5. PASO : Lista de variables

A = Variable de selección (Menú)

A$ = Variable de cadena (Sí/No)

I = Intensidad

R = Resistencia

U = Diferencia de potencial

6. PASO : Descripción del programa

Línea 10 - 60	Título y aclaraciones
Línea 70 - 100	Visualización del menú
Línea 110	Selección del menú
Línea 120 - 130	Sí A es mayor que 1 (A = Variable de menú) proseguir en la línea 190 una vez borrada la pantalla.
Línea 140 - 150	Introducciones de acuerdo con A = 1
Línea 160 - 170	Cálculos y visualización de resultados
Línea 180	Bifurcación a línea 300
Línea 190 - 240	Cálculos iguales a los anteriores pero para A = 2

Línea 250 - 280 Cálculo análogo pero para A = 3

Línea 300 - 310 Consulta si se desean más cálculos

en caso afirmativo, retorna a 70 tras borrar la pantalla

Línea 320 Final del programa

7. PASO : Resultados

Si seleccionamos el punto 1 del menú, el programa nos pedirá la diferencia de potencial y la intensidad. Si en respuesta introducimos por ejemplo el valor 220 para la diferencia de potencial y el valor 11 para la intensidad, obtendremos el resultado siguiente:

```
RESISTENCIA      : 20


OTRO CALCULO (S/N)?
```



Capítulo 5 : Lenguas

5.1 Consideración previa y más BASIC

Los ordenadores pueden cumplir también útiles servicios para el aprendizaje y el repaso de idiomas extranjeros. No se trata aquí de presentar cursos completos de idiomas (por ejemplo, para aprender inglés), porque esto superaría ampliamente el marco y el nivel de este libro - además, estos "programas completos" pueden adquirirse ya en los comercios a precios relativamente asequibles. Nos interesa mucho más el confeccionar programas sencillos con pocos medios, que puedan servirle de ayuda al escolar a la hora de aprender idiomas.

Se consideran aquí en primera línea aquellos programas que facilitan el aprendizaje de vocablos o que proporcionan traducciones, ahorrándonos así la consulta de diccionarios.

No obstante para este tipo de programas resultan necesarias algunas instrucciones BASIC nuevas.

Instrucción 20:

```
nn Nombre$ = INKEY$
```

Esta instrucción permite obtener información de la entrada de teclado que se almacena directamente en la variable de string Nombre$. Esta entrada no requiere pulsación de la tecla ENTER. En los siguientes programas explicaremos mejor su función.

A menudo nos será útil poder separar determinados segmentos del programa principal para poder bifurcar a los mismos más de una vez.
Estos programas parciales se denominan subrutinas. La bifurcación del programa principal a la subrutina requiere la instrucción siguiente:

Instrucción 21:

```
nn GOSUB mm
```

Esta instrucción provoca una bifurcación a la subrutina que comienza en la línea mm.

Una vez recorrida la subrutina debe retornarse al programa principal y cerrar la subrutina mediante la instrucción siguiente:

Instrucción 22:

```
nn RETURN
```

Esta instrucción provoca un salto de vuelta a aquel punto del programa principal situado justo después de la correspondiente instrucción GOSUB.

El funcionamiento de ambas instrucciones queda patente en los siguientes programas, de modo que prescindiremos de cualquier ejemplo.

5.2 Verbos irregulares ingleses

1. PASO : Presentación del problema

En este primer ejemplo presentamos un programa que tras introducir un verbo en castellano ofrece la traducción al inglés y las correspondientes formas irregulares.

Así, por ejemplo tras introducir en el ordenador

ir

éste nos contestA

go went gone (ir).

Para que el programa no resulte demasiado largo, nos limitaremos a 20 verbos irregulares. El lector descubrirá de inmediato cómo debe completar el programa para obtener una panorÁmica sobre todos los verbos irregulares ingleses.

2. PASO : Análisis del problema

En este programa deben disponerse verbos castellanos, sus traducciones al inglés y las formas irregulares de estos verbos.

Cuando la cantidad de información es muy amplia, resulta aconsejable no leer las informaciones de partida tras cada comienzo del programa, sino efectuar la lectura interactiva de la información requerida a partir de un fichero grabado en una diskette convenientemente preparada.
Esta posibilidad, que seguramente utilizarán los programadores más adelantados - por lo menos si disponen de una unidad de diskettes - no vamos a tratarla aquí.

Así, tras cada inicio del programa haremos que el usuario indique el verbo acerca del cual quiere información. A renglón seguido, el programa buscará en el archivo de datos. Una vez hallado el verbo, podrá visualizarse el correspondiente juego de datos.

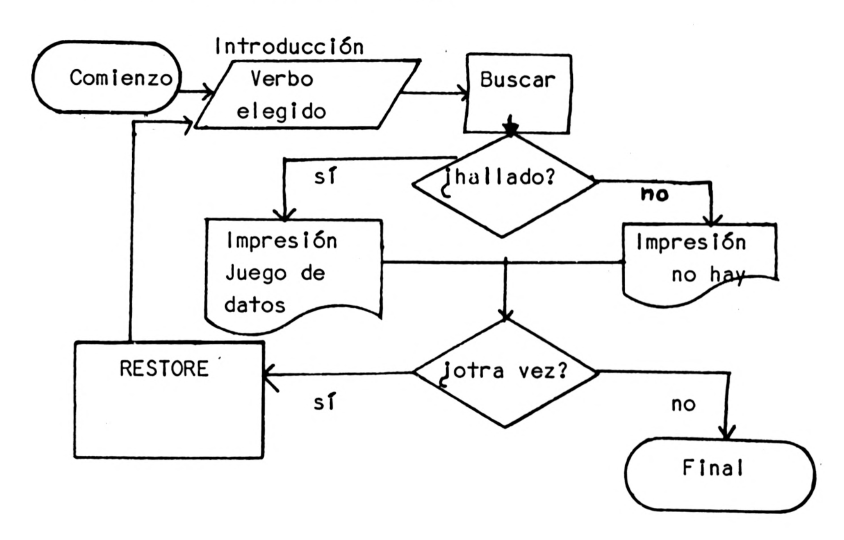
Si el programa, al comparar, no hallase el verbo introducido en su archivo de datos, podría deberse a que

- a que el verbo no existe aún en el archivo de datos (en principio solo hemos registrado 20 verbos
- a que el verbo introducido no es un verbo irregular.

En ambos casos, que debido a que la información de partida no es completa no pueden separarse el uno del otro, deberá visualizarse un mensaje.

En otro estado de cosas, el programa debe ofrecerle al usuario la posibilidad de consultar otro verbo.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM L1-VERBOS IRREGULARES INGLESES
20 CLS
30 PRINT "PROGRAMA PARA LA IMPRESION DE VERBOS"
40 PRINT "IRREGULARES INGLESES"
60 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W.VOSS, 1984"
70 PRINT: PRINT: PRINT: PRINT
80 PRINT "EL NUMERO DE VERBOS EXISTENTES ESTA LIMITADO EN"
90 PRINT "ESTE PROGRAMA. PUEDEN EFECTUARSE AMPLIACIONES"
100 PRINT "PROLONGANDO LA LISTA DE DATAS "
110 PRINT "(LINEA 500 Y SIGUIENTES)"
120 PRINT "Y AUMENTANDO EN LINEA 160 EL VALOR N"
130 PRINT: PRINT: PRINT
135 LOCATE 5,23: PRINT "PULSE UNA TECLA POR FAVOR !"
140 A$ = INKEY$
150 IF A$ = "" THEN 140
155 CLS: PRINT TAB(12) "ESPERE POR FAVOR": PRINT: PRINT
160 N = 20
170 DIM W$(N,4)
180 FOR I = 1 TO N: FOR J = 1 TO 4: READ W$(I,J): NEXT J: NEXT I
190 CLS
200 PRINT "INTRODUCIR EL VERBO EN ESPANOL DEL CUAL SE"
210 PRINT "DESEAN SABER LAS FORMAS INGLESAS :"
220 INPUT "ENTRADA : "; V$
230 FOR I = 1 TO N
240 IF V$=W$(I,1) THEN 275
250 NEXT I
260 PRINT: PRINT: PRINT "VERBO NO EXISTENTE EN EL FICHERO "
270 PRINT "O BIEN NO ES IRREGULAR.": GOTO 310
275 PRINT: PRINT: PRINT
280 PRINT "VERBO : "; W$(I,1): PRINT: PRINT; PRINT
290 PRINT "FORMAS INGLESAS  : ": PRINT
300 FOR J = 2 TO 4: PRINT W$(I,J); "   ";: NEXT J
310 PRINT: PRINT: PRINT: INPUT OTRO VERBO (S/N) "; A$
```

```
320 IF A$ = "S" THEN CLS: GOTO 200
330 PRINT: PRINT: PRINT "FINAL": END
500 DATA SER,BE,WAS,BEEN
502 DATA HABER,HAVE,HAD,HAD
504 DATA COMENZAR,BEGIN,BEGAN,BEGUN
506 DATA TRAER,BRING,BROUGHT,BROUGHT
508 DATA VENIR,COME,CAME,COME
510 DATA HACER,DO,DID,DONE
512 DATA COMER,EAT,ATE,EATEN
514 DATA COMPRAR,BUY,BOUGHT,BOUGHT
516 DATA SENTIR,FEEL,FELT,FELT
518 DATA ENCONTRAR,FIND,FOUND,FOUND
520 DATA IR,GO,WENT,GONE
522 DATA PERDER,LOSE,LOST,LOST
524 DATA HACER,MAKE,MADE,MADE
526 DATA LEER,READ,READ,READ
528 DATA CORRER,RUN,RAN,RUN
530 DATA DECIR,SAY,SAID,SAID
532 DATA VER,SEE,SAW,SEEN
534 DATA SENTAR,SIT,SAT,SAT
536 DATA HABLAR,SPEAK,SPOKE,SPOKEN
538 DATA COGER,TAKE,TOOK,TAKEN
```

5. PASO : Lista de variables

A$ = Variable de cadena (sí/no)

I = Indice variable
J = Indice variable
N = Cantidad total de sentencias DATA

V$ = Verbo seleccionado
W$ = Verbo en castellano

6. PASO : Descripción del programa

Línea 10 - 135	Título, aclaraciones e indicación de la manera de actualizar el archivo de datos.
Línea 140 - 150	Programa en posición de 'espera'. La ejecución continúa después de pulsar una tecla.
Línea 155	Borrado de pantalla
Línea 160 - 180	Dimensionado y lectura de los datos
Línea 190 - 220	Borrado de la pantalla y demanda del verbo elegido
Línea 230 - 250	Búsqueda del verbo en el fichero de datos Si lo encuentra continúa el proceso en 275 Si no lo encuentra prosigue en 260
Línea 260 - 270	Mensaje indicando que el verbo no existe en el fichero de datos y bifurcación a línea 310
Línea 275 - 300	Visualización de la información buscada
Línea 310 - 320	Consulta si se desea otra operación, en caso afirmativo borrado de la pantalla y retorno a línea 200
Línea 330	En caso negativo finaliza el programa
Línea 500 - 538	Sentencias de datos

7. PASO : Resultados

Si introducimos por ejemplo en respuesta a la demanda del ordenador en la línea 200:

IR,

El ordenador contesta:

VERBO : IR

FORMAS INGLESAS :

GO
WENT
GONE

OTRA VEZ (S/N) ?

Puede verse claramente como hay que modificar el programa para completar la lista de verbos irregulares : Un la línea 500 y siguientes habrá que añadir más instrucciones DATA y en la línea 160 habrá que indicar el nuevo número de verbos.

5.3 Vocablos franceses

1. PASO : Presentación del problema

En este ejemplo vamos a mostrar como puede encontrar aplicación el ordenador para sustituir la libreta de vocablos. Este contesta a la palabra francesa que le introducimos con el equivalente castellano o viceversa (rige aquí la misma observación del punto 5.2 acerca de la utilización interactiva de una memoria de diskettes).

Para que el programa no crezca demasiado, lo limitaremos a diez vocablos. Es decir, que sirve sólo a modo de ejemplo. Para aprovecharlo realmente, el lector deberá tener la paciencia de introducir previamente en el ordenador todo su archivo castellano-francés.

En los principios de funcionamiento del programa no hay ningún cambio.

2. PASO : Análisis del problema

Con este problema podemos ser escuetos:

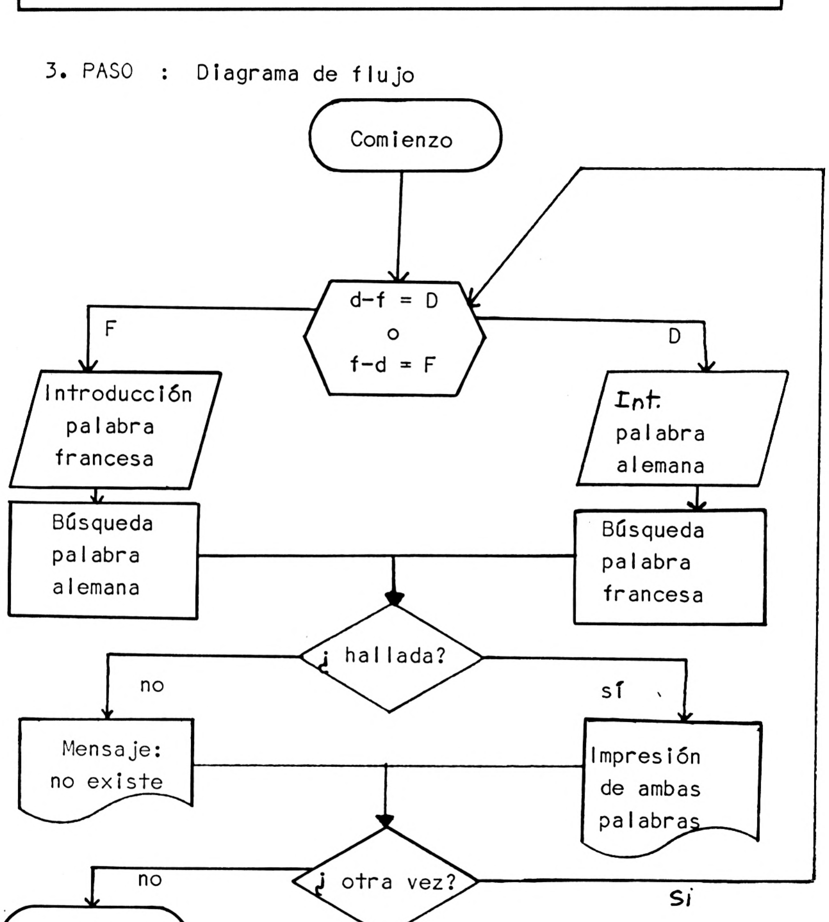
Cuando el usuario introduce una palabra castellana, el ordenador debe buscar entre su archivo de datos el correspondiente par de palabras castellano-francesas mediante la comparación con todos los vocablos castellanos (y sus correspondientes parejas en francés) de que dispone para, después, visualizarlas.

Para que ambos "sentidos" de la traducción puedan ser cubiertos por un solo programa, primero hay que decirle al ordenador si se desea la traducción del castellano al francés o del francés al castellano.

En caso de no poder traducir la palabra introducida por no estar incluida (aún) en la (muy) corta lista de vocablos, deberá visualizarse el correspondiente mensaje.

Además, debe ofrecérsele al usuario la posibilidad de obtener otra traducción en la ejecución del mismo programa, si así lo desea.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM L2-VOCABLOS CASTELLANO-FRANCES
20 CLS
30 PRINT "PROGRAMA PARA CONSULTAR VOCABLOS"
40 PRINT: PRINT "CASTELLANO-FRANCES O VICEVERSA"
50 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W.VOSS, 1984": PRINT
60 PRINT: PRINT: PRINT: PRINT
70 PRINT "ESTE  PROGRAMA CONTIENE UNICAMENTE "
80 PRINT " 10 VOCABLOS EN SU FICHERO": PRINT
90 PRINT "PARA AMPLIAR EL PROGRAMA DEBE AGREGAR "
100 PRINT "SENTENCIAS DATA A PARTIR DE LA LINEA 500"
120 PRINT "Y MODIFICAR EL VALOR N EN LINEA 160"
130 LOCATE 5,23: PRINT "PULSE UNA TECLA POR FAVOR"
140 A$ = INKEY$: IF A$ = "" THEN 140
150 CLS: PRINT "ESPERE POR FAVOR"
160 N = 10
170 DIM D$(N),F$(N)
180 FOR I = 1 TO N: READ D$(I),F$(I): NEXT I
190 CLS
200 PRINT "QUE TRADUCCION DESEA? "
210 PRINT: PRINT: PRINT "-  CASTELLANO-FRANCES   (1) "
220 PRINT: PRINT "O BIEN"
230 PRINT: PRINT "-   FRANCES-CASTELLANO   (2) "
240 PRINT: PRINT: INPUT "PULSE 1 O 2, POR FAVOR: "; Z
245 CLS
250 IF Z = 2 THEN 350
260 PRINT: PRINT: INPUT "VOCABLO EN CASTELLANO: ";W$
280 FOR I = 1 TO N: IF W$=D$(I) THEN 320
290 NEXT I
300 PRINT: PRINT "VOCABLO NO ENCONTRADO!"
310 PRINT "EN EL FICHERO": GOTO 450
320 CLS:   PRINT "CASTELLANO  : "; D$(I)
330 PRINT: PRINT "FRANCES     : "; F$(I)
```

```
340 GOTO 450
350 PRINT: PRINT: INPUT "VOCABLO FRANCES POR FAVOR : ";W$
360 FOR I = 1 TO N: IF W$ = F$(I) THEN 400
370 NEXT I
380 PRINT: PRINT  "VOCABLO NO ENCONTRADO !"
390 PRINT "EN EL FICHERO": GOTO 450
400 CLS: PRINT   "FRANCES      : "; F$(I)
410 PRINT: PRINT "CASTELLANO   : "; D$(I)
450 PRINT: PRINT: PRINT: INPUT "OTRO VOCABLO (S/N)? ";A$
460 IF A$ = "S" THEN CLS: GOTO 200
470 PRINT: PRINT "FINAL DEL PROGRAMA": END
500 DATA COMPRAR,ACHETER
502 DATA HABER,AVOIR
504 DATA IR,ALLER
506 DATA VER,VOIR
508 DATA BEBER,BOIRE
510 DATA HACER,FAIRE
512 DATA LEER,LIRE
514 DATA COMER,MANGER
516 DATA TOMAR,PRENDRE
518 DATA FUMAR,FUMER
```

5. PASO : Lista de variables

A$	Variable de string (Sí/No)
D$	Vocablo en castellano
F$	Vocablo en francés
I	Indice variable
N	Cantidad de sentencias de datos
W$	Palabra seleccionada
Z	Dígito de selección del menú

6. PASO : Descripción del programa

Línea 10 - 120	Títulos, aclaraciones, etc.
Línea 130 - 150	Borrado de la pantalla y pausa
Línea 160 - 180	Indicación de la cantidad de líneas de datos y dimensionado de variables.
Línea 190 - 240	Borrado de la pantalla y visualización del menú de selección
Línea 245 - 250	Si se ha seleccionado 2 (Francés-Castellano) la ejecución continúa en línea 350
Línea 260	Introducción de un vocablo castellano
Línea 280 - 290	Búsqueda del vocablo en el fichero de datos
Línea 300 - 310	Si el vocablo no se encuentra aún en el fichero, bifurca a línea 450
Línea 320 - 330	Visualización de los resultados
Línea 340	Bifurcación a línea 450
Línea 350 - 410	Procedimiento análogo para introducir vocablos en francés
Línea 450 - 470	Finalización del programa con alternativa de proseguir con un nuevo vocablo
Línea 500 - 518	Sentencias de datos

7. PASO : Resultados.

Si pulsamos el número 1 en el menú, decidiéndonos por la traducción del castellano al francés, el ordenador nos pedirá una palabra castellana.

Si en respuesta introducimos la palabra, el ordenador imprimirá:

CASTELLANO : TENER

FRANCES : AVOIR

OTRA VEZ (S/N) ?

5.4 Test de vocablos ingleses

1. PASO : Presentación del problema

De forma similar al punto anterior, aquí deben ponerse a nuestra disposción traducciones apartir de un determinado fichero de vocablos. Pero, en conZraposición al programa precedente, el usuario es requerido para contestar a una palabra castellana elegida al azar por el ordenador con la palabra inglesa correcta.

Es decir, que presentamos un programa que funciona del mismo modo que el aprendizaje de vocablos con la columna de palabras en el idioma extranjero tapada de nataño.

Puede observarse rápidamente que este programa es también aplicable a otros idiomas.

Con el fin de reducir el tiempo de programación nos limitaremos a ofrecer solo diez vocablos a modo de ejemplo. El ordenador selecciona en cada caso un subconjunto aleatorio del mismo, siendo informado el usuario acerca del número de aciertos por serie.

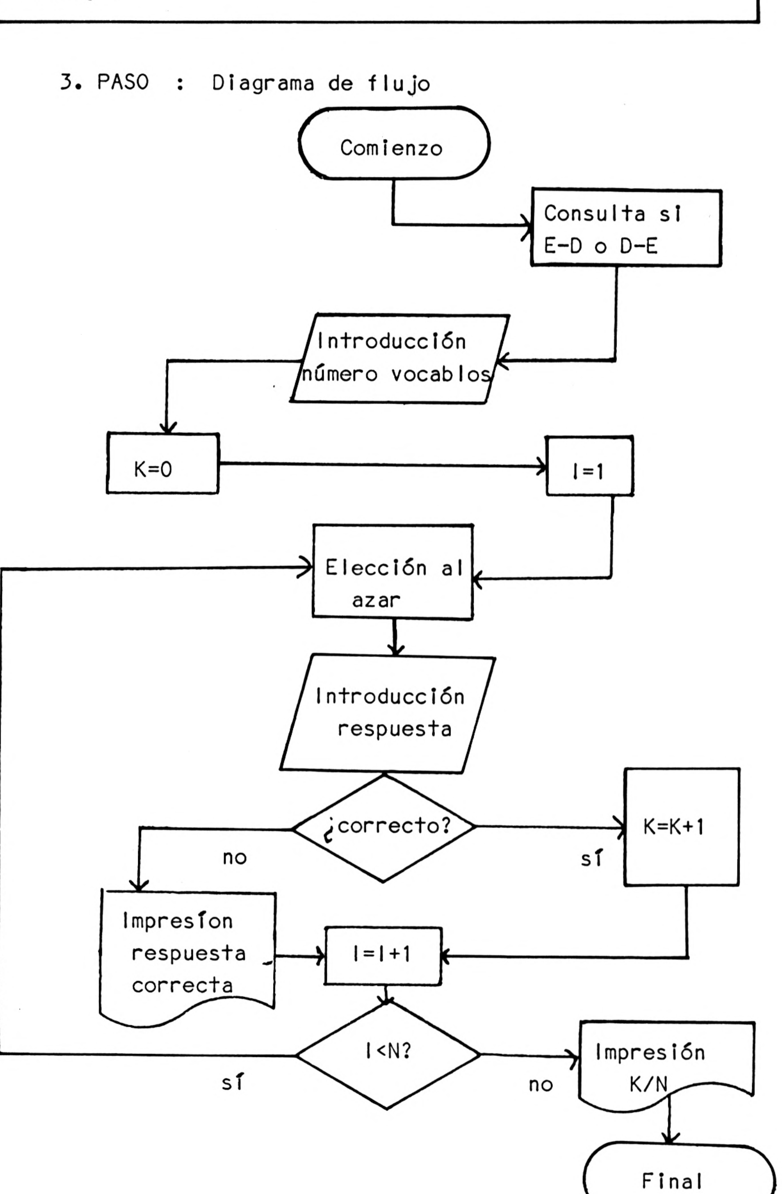
2. PASO : Análisis del problema

Partiendo de un fichero de datos dado, el programa debe realizar una seleción al azar.

El usuario responde a cada vocablo visualizado. El número de respuestas correctas es comparado con el número de preguntas.

El análisis del problema no ofrece ninguna otra dificultad.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM L3-TEST DE VOCABLOS
20 CLS
30 PRINT "PROGRAMA PARA CONSULTAR VOCABLOS"
40 PRINT: PRINT "AQUI: CASTELLANO/INGLES O VICEVERSA"
50 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS, 1984"
60 PRINT: PRINT: PRINT: PRINT
70 PRINT "EL PRESENTE PROGRAMA CONTIENE UNICAMENTE"
80 PRINT "10 VOCABLOS EN SU FICHERO DE DATOS":PRINT
90 PRINT "PARA AMPLIARLO ES NECESARIO AGREGAR"
100 PRINT "SENTENCIAS DATA A PARTIR DE LA LINEA 1000"
120 PRINT "Y MODIFICAR EL VALOR DE N EN LA LINEA 160"
130 LOCATE 5,23: PRINT "PULSE UNA TECLA, POR FAVOR! "
140 A$=INKEY$: IF A$="" THEN 140
150 CLS: PRINT "ESPERE UN MOMENTO, POR FAVOR!"
160 N = 10
170 DIM D$(N,2)
180 FOR I = 1 TO N: READ D$(I,1),D$(I,2): NEXT I
185 ZZ = 1
190 CLS
200 PRINT "CASTELLANO/INGLES      (1) "
210 PRINT: PRINT: PRINT "O BIEN"
220 PRINT "INGLES/CASTELLANO      (2) "
230 PRINT: PRINT: PRINT "PULSE 1 O 2, POR FAVOR ": PRINT
240 INPUT "                           "; Z
245 IF Z<>ZZ THEN GOSUB 3000: REM CONSULTA
250 PRINT: PRINT: PRINT "CUANTOS VOCABLOS QUIERE CONSULTAR"
260 INPUT " "; A
270 GOSUB 2000
400 PRINT: INPUT "OTRA VEZ (S/N)? "; A$
410 IF A$ = "S" THEN CLS: ZZ = Z: GOTO 200
420 PRINT: PRINT "FINAL DEL PROGRAMA": END
450 PRINT: PRINT: PRINT: INPUT "NUEVAMENTE (S/N)? "; A$
460 IF A$ = "S" THEN CLS: GOTO 200
```

```
470 PRINT: PRINT "FINAL DEL PROGRAMA: END
1000 DATA CORRER,RUN,ESCRIBIR,WRITE
1010 DATA IMPRIMIR,PRINT, IR,GO
1020 DATA SI, IF SUBRUTINA,SUBROUTINE
1030 DATA ORDENADOR,COMPUTER,CARACTER,CHARACTER
1040 DATA PANTALLA,SCREEN,TECLA,KEY
2000 CLS
2005 K=0
2010 I=1
2020 R=INT(RND(1)*N+1)
2030 PRINT D$(R,1);: INPUT "   RESPUESTA  : ";W$
2040 IF W$=D$(R,2) THEN K=K+1: PRINT: PRINT: PRINT TAB(10) "MUY
     BIEN! ": GOTO 2070
2050 PRINT: PRINT: PRINT "E R R O R"
2060 PRINT: PRINT "LA PALABRA CORRECTA ES : "; D$(R,2)
2070 PRINT: PRINT: PRINT: I=I+1
2080 IF I<=A THEN 2020
2090 KA=(K/A)*100
2100 PRINT: PRINT "PORCENTAJE DE RESPUESTAS CORRECTAS :"; KA;" %"
2110 RETURN
3000 REM INTERCAMBIO
3010 FOR I=1 TO N
3020 H$=D$(I,1): D$(I,1)=D$(I,2): D$(I,2)=H$
3030 NEXT I
3040 RETURN
```

5. PASO : Lista de variables

A	= Número de vocablos a comprobar
A$	= Cadena de respuesta (S/N)
D$	= Vocablos castellanos
H$	= Campo auxiliar para intercambio
I	= Indice variable
K	= Número de respuestas correctas
KA	= Porcentaje de respuestas correctas

N = Número total de vocablos del fichero
R = Número aleatorio
W$ = Respuesta del usuario
Z = Opción escogida del menú
Z = Valor auxiliar (inicial 1), para el cambio eventual de castellano/inglés

6. PASO : Descripción del programa

Línea 10 - 120	Título, aclaraciones y recordatorio de la eventual ampliación de los datos del programa.
Línea 130 - 150	Borrado de la pantalla y pausa
Línea 160 - 180	Indicación del número de vocablos incluídos en el fichero de datos, dimensionado y lectura.
Línea 185	Inicialización (1) de la variable auxiliar Z
Línea 190 - 240	Borrado de la pantalla, visualización del menú y demanda de la selección.
Línea 245	Consulta si se ha elegido la opción 1 En caso afirmativo, continúa en línea 250 de lo contrario bifurca a línea 3000
Línea 250 - 260	Solicita el número de vocablos a consultar

Línea 270	Bifurca a línea 2000
Línea 400 - 420	Interroga si se desea continuar (retorno a línea 200) y finaliza el programa
Línea 1000 - 1020	Datos
Línea 2000 - 2110	UP Elección al azar
2005 - 2010	Se inicializan contadores K e I
2020	Lectura número aleatorio
2030	Visualización del vocablo y demanda de respuesta
2040	Si la respuesta es correcta bifurcación a línea 2070
2050 - 2060	Mensaje de error y visualización de la respuesta correcta
2070 - 2080	Vocablo siguiente
2090 - 2100	Visualización de la cuota de aciertos
2110	Retorno al programa principal
Línea 3000 - 3040	UP Cambio: ambio Castellano/inglés - inglés/castellano

7. PASO : Resultados

No consideramos necesario ofrecer aquí muestra de los resultados. Aquellos que se visualizan durante el transcurso del programa resultan directamente de la descripción del mismo (concretamente ver líneas 2040 hasta 2100)

5.5 La clasificación de vocablos

1. PASO : Presentación del problema

La clasificación de datos es uno de los problemas más importantes que puede resolver un ordenador. Pero no sólo es importante la posibilidad de mantener el orden entre datos númericos, sino también la clasificación de datos alfabéticos o de string de caracteres.

Existen un gran número de algoritmos diferentes para lograr la clasificación de datos, que en último término se diferencian especialmente en la velocidad del proceso.

Pero no es este aspecto al que nos queremos referir ahora, en este momento nos interesa presentar un algoritmo que expresa claramente las bases fundamentales del proceso de clasificación de datos.

2. PASO : Análisis del problema

"Clasificar" significa la comparación reiterada de 2 strings del conjunto de datos:

Comparamos el primer string paso a paso con todos los restantes. Si el primer string sigue un orden alfabético superior al segundo, se deben intercambiar los contenidos de ambos.

De esta forma sabemos que una vez comparado el primer string con todos los restantes (en el primer bucle), la primera posición la ocupe el primer string del orden alfabético.

A continuación se compara el string de la segunda posición con todos los restantes (excepto el primero). También aquí se intercambian los valores que correspondan. Al final del segundo bucle de comparación sabemos que en la segunda posición se encuentra el string que realmente sigue el segundo lugar del orden alfabético.

Seguidamente comparamos el string de la tercera posición con todos los restantes ... etc.

Se ejecutan en total tantos bucles como posibilidades de comparación existen.

Si por ejemplo deben clasificarse 5 strings, se ejecutan 4 bucles: en el primer bucle hay 4 comparaciones, en el segundo 3, en el tercero 2 y en el cuarto 1 comparación.

Visión esquemática del proceso:

1. Bucle : Comparación campo 1 con campo 2
Comparación campo 1 con campo 3
Comparación campo 1 con campo 4
Comparación campo 1 con campo 5

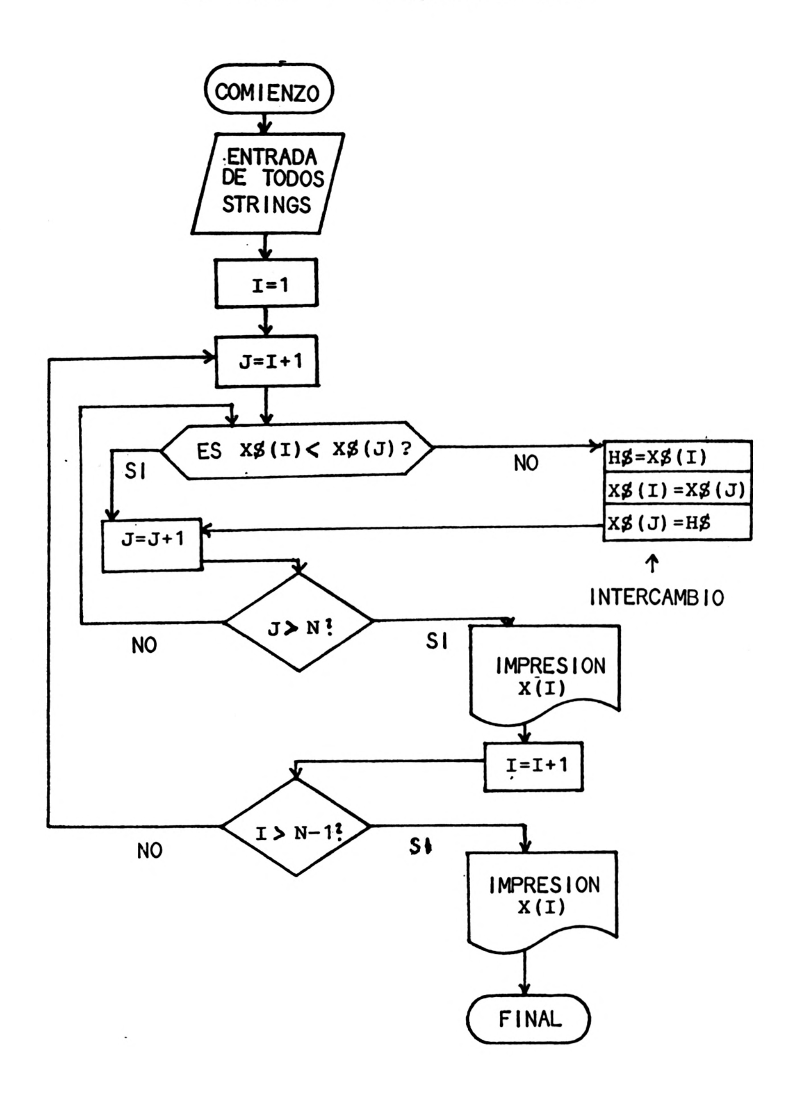
2. Bucle : Comparación campo 2 con campo 3
Comparación campo 2 con campo 4
Comparación campo 2 con campo 5

3. Bucle : Comparación campo 3 con campo 4
Comparación campo 3 con campo 5

4. Bucle : Comparación campo 4 con campo 5

Se debe tener en cuenta que después de las primeras comparaciones, los contenidos de los campos siempre pueden ser otros, según las posiciones en las que se han realizado los intercambios.

3. PASO : Diagrama de flujo



4. PASO : Programa

A diferencia del procedimiento previsto por el diagrama de flujo anterior, en este programa se prevee adicionalmente que el usuario pueda determinar la clasificación de los vocablos alfabéticamente, según la ortografía castellana o inglesa.

```
10 REM L4-CLASIFICACION
20 CLS
30 PRINT TAB(2) "PROGRAMA PARA LA CLASIFICACION DE VOCABLOS"
40 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS, 1984"
50 PRINT: PRINT: PRINT: PRINT
60 PRINT "EN ESTE PROGRAMA SE CLASIFICAN UNICAMENTE"
70 PRINT "10 VOCABLOS INGLESES": PRINT
80 PRINT "SI SE QUIEREN CLASIFICAR OTROS VOCABLOS"
90 PRINT "SE DEBEN MODIFICAR LAS SENTENCIAS DATAS DESDE "
100 PRINT " LINEA 500 Y ALTERAR EL VALOR DE N EN LINEA 150"
110 LOCATE 5,23
120 PRINT "PULSE UNA TECLA POR FAVOR"
130 A$=INKEY$: IF A$="" THEN 130
140 CLS
150 N=10
160 DIM E$(N),D$(N)
170 FOR I=1 TO N: READ E$(I),D$(I): NEXT I
180 PRINT "SE DESEA CLASIFICAR LOS VOCABLOS INGLESES "
190 PRINT "O BIEN LOS VOCABLOS CASTELLANOS"
200 PRINT: INPUT "INTRODUZCA 'I' O BIEN 'C' : ";A$
210 IF A$="C" THEN 240
220 CLS: PRINT "INGLES"; TAB(20) "CASTELLANO": PRINT: PRINT
230 GOSUB 1000 : REM A CLASIFICAR
235 GOTO 270
240 FOR I=1 TO N: H$=E$(I): E$(I)=D$(I): D$(I)=H$: NEXT I
250 CLS: PRINT "CASTELLANO"; TAB(20) "INGLES": PRINT: PRINT
260 GOSUB 1000: REM A CLASIFICAR
```

```
270 PRINT: PRINT "FINAL DEL PROCESO": END
500 DATA PRINT,IMPRIMIR,WRITE,ESCRIBIR,END,FIN,GO,IR
510 DATA RUN,CORRER,DATA,DATOS,SUBROUTINE,SUBRUTINA
520 DATA INPUT,ENTRADA,SAVE,ALMACENAR,LOAD,CARGAR
1000 REM CLASIFICA
1010 FOR I=1 TO N-1
1020 FOR J=I+1 TO N
1030 IF E$(I)<=E$(J) THEN 1070
1040 H$=E$(I): G$=D$(I)
1050 E$(I)=E$(J): D$(I)=D$(J)
1060 E$(J)=H$: D$(J)=G$
1070 NEXT J
1080 PRINT E$(I); TAB(20) D$(I)
1090 NEXT I
1100 PRINT E$(I); TAB(20) D$(I)
1110 RETURN
```

5. PASO : Lista de variables

A$ = Cadena para respuestas (S/N)

D$ = Vocablos castellanos
E$ = Vocablos ingleses
G$ = Campo auxiliar 1 para intercambios
H$ = Campo auxiliar 2 para intercambios

I = Indice variable
J = Indice variable
N = Número total de vocablos a tratar

6. PASO : Descripción del programa

Línea 10 - 110	Cabeceras y aclaraciones
Línea 120 - 140	Borrado de pantalla y pausa
Línea 150 - 170	Determinación del número de vocablos a tratar, dimensionado y lectura de datos
Línea 180 - 210	Interroga si se desea la clasificación por los vocablos castellanos o ingleses.
Línea 220	Visualización de la tabla
Línea 230	Bifurcación a línea 1000 para clasificar
Línea 235	Bifurcación a línea 270
Línea 240	Intercambio de vocablos castellanos e ingleses
Línea 250	Visualización de la tabla
Línea 260	Bifurcación a línea 1000 para clasificar
Línea 270	Finalización del programa
Línea 500 - 520	Datos

Línea 1000 - 1110	Clasificación:
	Subrutina para la clasificación de strings
1010	Contador de bucles
1020	Contador de comparación
1030	Comparación Si resulta satisfecho, sigue en línea 1070 de lo contrario en 1040
1040-1060	Intercambio
1070	Siguiente comparación
1080	Ya que no pueden realizarse más comparaciones se visualizan los 2 strings actualmente en primera posición
1090	Siguiente bucle
1100	Visualización de la última pareja de strings
1110	Retorno

7. PASO : Resultados

Supongamos que se na efectuado la clasificación de los vocablos castellanos:

CASTELLANO	INGLES
CONSEGUIR	GET
DATOS	DATA
FIN	END
IMPRIMIR	PRINT
IR	GO
.....	......



Capítulo 6 : Biología/Ecología

6.1 Aspectos previos

La aplicación del ordenador para problemas biológico-ecológicos se refiere especialmente a la simulación de procesos de desarrollo y de crecimiento (naturalmente pueden confeccionarse también programas de consulta siguiendo el patrón de los del capítulo anterior). Estos desarrollos pueden visualizarse en la pantalla gráficamente o en forma de tabla.

6.2 Crecimiento exponencial

1. PASO : Presentación del problema

Una población inicial de X0 individuos tiene una tasa de crecimiento anual del R por ciento (R debe ser positivo). Si este crecimiento no choca con limitaciones externas - cosa muy improbable a largo plazo - se produce un llamado crecimiento exponencial.

El desarrollo de un proceso de crecimiento tan sencillo como éste, se simulará con un programa BASIC.

2. PASO : Análisis del problema

Para este programa hay que introducir la población inicial y la tasa de crecimiento anual.

Después de un año se obtiene la población X1:

X1 = X0 + X0*R/100 = X0*(1 + R/100)

Después de dos años se obtiene una X2:

X2 = X1*(1 + R/100) = X0*(1 + R/100)*(1 + R/100)

De lo que resulta:

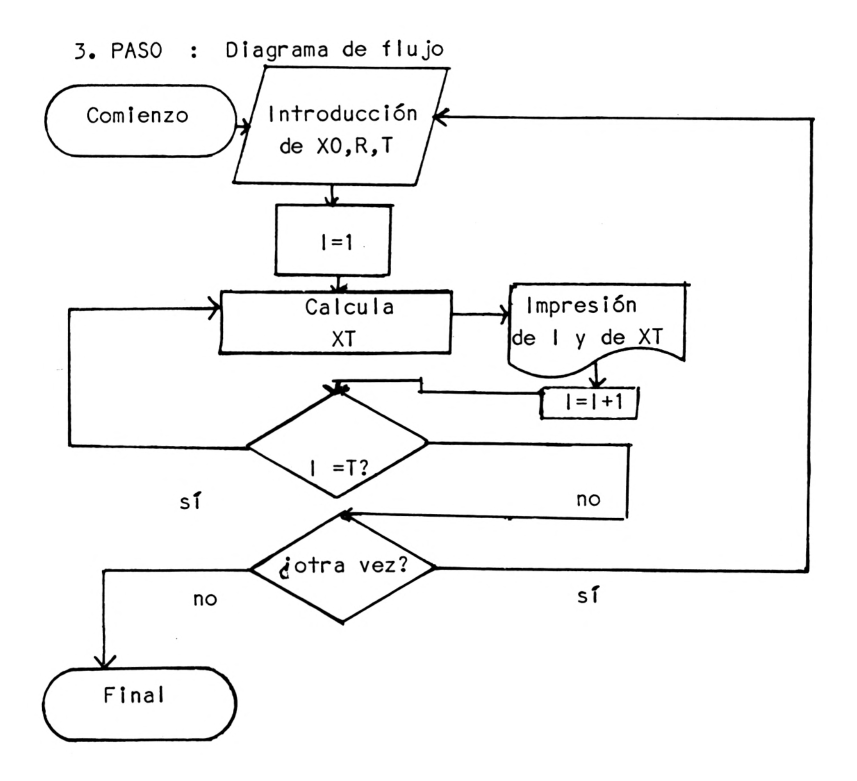
$$X2 = X0 * (1 + R/100)^{2}$$

Generalizando, después de T años se cumple que:

$$XT = X0 * (1 + R/100)^{T}$$

Esta fórmula indica el algoritmo de cálculo para el correspondiente programa BASIC.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM B1-CRECIMIENTO
20 CLS
30 PRINT "PROGRAMA  PARA REPRESENTAR EN FORMA DE"
40 PRINT "TABLA EL CRECIMIENTO EXPONENCIAL"
50 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS, 1984"
60 PRINT: PRINT: PRINT: PRINT
70 INPUT         "POBLACION INICIAL           : "; X0
80 PRINT: INPUT "TASA DE CRECIMIENTO EN %   : "; R
90 PRINT: INPUT "NUMERO DE ANOS              : "; T
100 CLS: PRINT "TRAS .... ANOS    POBLACION": PRINT: PRINT
110 I=1
120 XT=X0*(1+R/100)^I
130 PRINT TAB(5)I; TAB(19) USING "≠≠≠≠≠.≠≠"; XT
140 I=I+1
145 IF I/10 = INT(I/10) THEN PRINT:  PRINT "PULSE CONT POR FAVOR":
    STOP
150 IF I<=T THEN 120
160 PRINT: PRINT: INPUT "OTRA VEZ (S/N) "; A$
170 IF A$="S" THEN CLS: GOTO 70
180 PRINT: PRINT: PRINT "FINAL DEL PROGRAMA": END
```

5. PASO : Lista de variables

A$ = String de respuesta
I = Indice variable
R = Tasa de crecimiento en %
T = Tiempo en años
X0 = Población inicial
XT = Población en el momento T

6. PASO : Descripción del programa

Línea 10 - 60	Cabeceras
Línea 70 - 90	Visualización de las informaciones INPUT que necesita el programa
Línea 100	Visualización de cabecera de la tabla
Línea 110	Predefinición del primer período
Línea 120	Cálculo de la población
Línea 130	Visualización
Línea 140	Paso al siguiente período
Línea 145	Después de cada 10 períodos se interrumpe el programa (es decir cuando I es múltiplo de 10); la ejecución continúa pulsando la tecla CONT, como viene indicado.
Línea 150	Mientras I sea menor o igual a T el cálculo debe continuar (línea 120).
Línea 160 - 180	Finalización del programa, excepto que se desee realizar otro proceso (entonces retorno a línea 70 después de borrar la pantalla)

7. PASO : Resultados

Si por ejemplo introducimos el valor 500 para la población inicial y una tasa de crecimiento del 4(%), además de un período de valor 8 (p.e. años), el programa genera la siguiente impresión en pantalla:

```
TRAS ... AÑOS  POBLACION

      1           520
      2           540.8
      3           562.43
      4           584.93
      5           608.33
      6           632.66
      7           657.97
      8           684.28

OTRA VEZ   (S/N)  ?
```

6.3 Crecimiento limitado

1. PASO : Presentación del problema

El crecimiento exponencial que ya presentamos en el punto anterior, no se da en la práctica, y si se da será por espacios de tiempo muy cortos. Deben considerarse ante todo efectos inhibidores, que serán más acentuados cuanto mayores sean los valores de las variables en juego.

Pensemos por ejemplo que una población con crecimiento exponencial choca muy pronto con limitaciones de nutrición, que dificultan más y más su posterior desarrollo.

Este tipo de evolución "amortiguada" va a ser simulada en el próximo programa BASIC.

2. PASO : Análisis del problema

El análisis del problema lleva consigo una forma de resolución similar a la del punto anterior:

También aquí se requieren valores de partida:

X0 = Población de partida
R0 = Tasa de crecimiento anual (en %) al comienzo del proceso de desarrollo

En contraposición al ejemplo precedente, no partimos de la base de que la tasa de crecimiento es constante, sino que provocamos su reducción a medida que aumenta la población.

Como la población aumenta con el paso del tiempo, mediante un esquema mental muy sencillo podemos deducir que la tasa de crecimiento se reduce progresivamente. Por ejemplo, podemos dividir entre dos la tasa cuando el tiempo se ha duplicado, dividirla entre cuatro cuando el tiempo se ha cuatriplicado, etc.

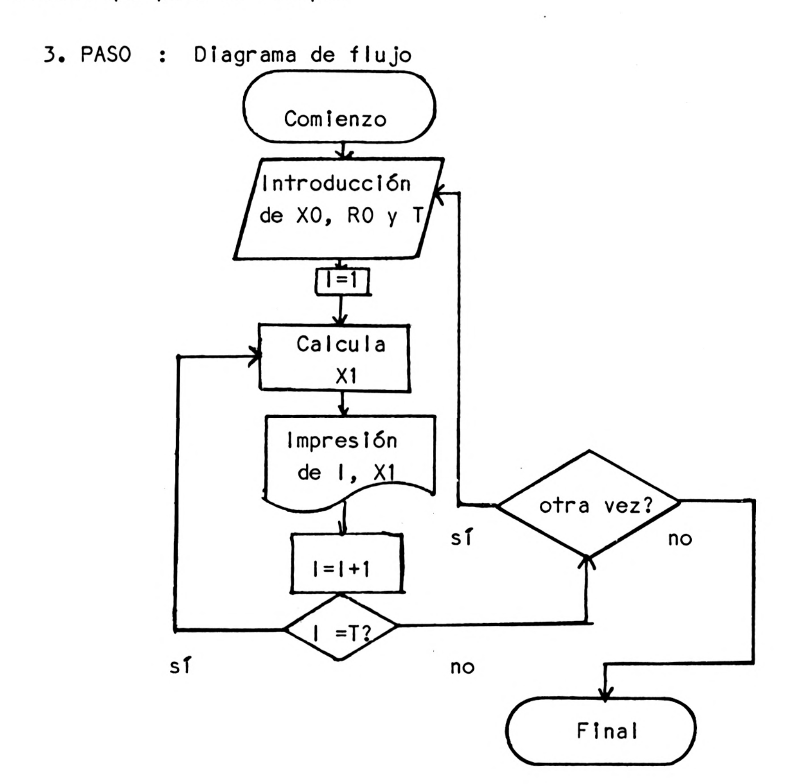
Claro que podemos establecer otros modelos matemáticos para simular este proceso de frenado, aunque esto no es de vital importancia dentro de este contexto.

El algoritmo que acabamos de describir puede expresarse matemáticamente de la forma siguiente:

X1 = X0 * (1 + R/I)

Es decir, en el período X1 la población se obtiene a partir de la del período precedente, como en el ejemplo anterior; en cambio, la tasa de crecimiento se divide entre el índice variable I, de manera que aquélla se reduce a medida que aumenta I, es decir, a medida que pasa el tiempo.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM B2-CRECIMIENTO LIMITADO
20 CLS
30 PRINT "PROGRAMA PARA SIMULAR EN UNA TABLA"
40 PRINT "UN PROCESO DE CRECIMIENTO FRENADO"
50 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS, 1984"
60 PRINT: PRINT: PRINT: PRINT: PRINT
70 INPUT "POBLACION INICIAL     : "; X0
80 PRINT: INPUT "TASA DE CRECIMIENTO DEL PRIMER ANO (%) : "; R
85 PRINT: INPUT "CUANTOS ANOS      : "; T
95 CLS
100 PRINT "TRAS .... ANOS    POBLACION    CRECIMIENTO (%)"
105 PRINT: PRINT
110 I = 1
120 X1 = X0 + X0 * (R/I) / 100
140 Z = (X1-X0) / X0: Z = Z * 100
150 PRINT (TAB) (5) I; TAB(19) USING "≠≠≠≠≠.≠≠";
    X1;: PRINT TAB(28) USING "≠≠≠≠.≠≠"; Z
160 X0=X1: I=I+1 : IF I<=T THEN 120
180 PRINT: INPUT "OTRA VEZ (S/N)?  "; A$
190 IF A$ = "S" THEN CLS: GOTO 70
200 PRINT: PRINT "FIN DE VISUALIZACION": END
```

5. PASO : Lista de variables

A$ = Cadena de respuesta (S/N)
I = Indice variable (número de períodos)
R = Tasa de crecimiento inicial
T = Tiempo en años
X0 = Población inicial
X1 = Población después de un período
Z = Modificación de la población

6. PASO : Descripción del programa

Línea 10-60 : Título y aclaraciones

Línea 70-95 : Demanda de los datos de partida y borrado de la pantalla

Línea 100-105 : Impresión del título de la tabla

Línea 110 : Fijación del contador de períodos

Línea 120-150 : Cálculo de la población, del crecimiento e impresión de los valores tras redondear a dos posiciones decimales

Línea 160 : X1 toma el valor de X0 y continuación de los cálculos mientras I siga siendo más pequeño que T después de elevarlo en 1

Línea 180-200 : Consulta si todavía se desea calcular (en caso afirmativo vuelta a 70) y final del programa

7. PASO : Resultados

Si introducimos como población inicial el valor 500, y una tasa de crecimiento en el primer año del 4%, además de indicar que el proceso debe durar 8 años, obtendremos:

TRAS...AÑOS	POBLACION	CRECIMIENTO (%)
1	520	4
2	530.4	2
3	537.47	1.33
4	542.84	1
5	547.18	.8
6	550.83	.67
7	553.98	.57
8	556.75	.5

OTRA VEZ (S/N) ?

6.4 Contaminación ambiental

1. PASO : Presentación del problema

Las poblaciones en crecimiento, por regla general frenan ellas mismas su desarrollo, debido a que deterioran sus condiciones de vida en grado creciente.

Un ejemplo típico de ello es la contaminación ambiental creciente a lo largo de las diversas fases de industialización.

Es decir, aquí también se produce un "frenado del crecimiento" - similar al del punto anterior - que modifica el crecimiento exponencial que conocemos del punto 6.2.

En contraposición al punto anterior, este "frenado" no opera de acuerdo con el tamaño de la población propiamente dicho, sino de forma indirecta en dependencia de una variable que, a su vez, es proporcional a la población.

2. PASO : Análisis del problema

En el primer período, la población resultante del período 0 ya no se calcula

X1 = X0 + X0 * R/100

sino:

X1 = X0 + X0 * (R/100 - P*B0)

B0 es el factor contaminación ambiental en el momento 0, que frena el crecimiento de la población por medio de un parámetro P adecuado (p.e. dentro del ámbito de P = 0.01).

Lógicamente, necesitamos también una relación funcional que pueda expresar B1 (contaminación ambiental en el período 1) en dependencia de B0 y de X0.

Un posible ejemplo de relación sería:

B1 = B0 + A * X0

Partiendo de un factor B0 = 0, la contaminación aumenta a medida que crece la población, de acuerdo con un parámetro A, que debe dimensionarse adecuadamente (p.e. A = 0.015).

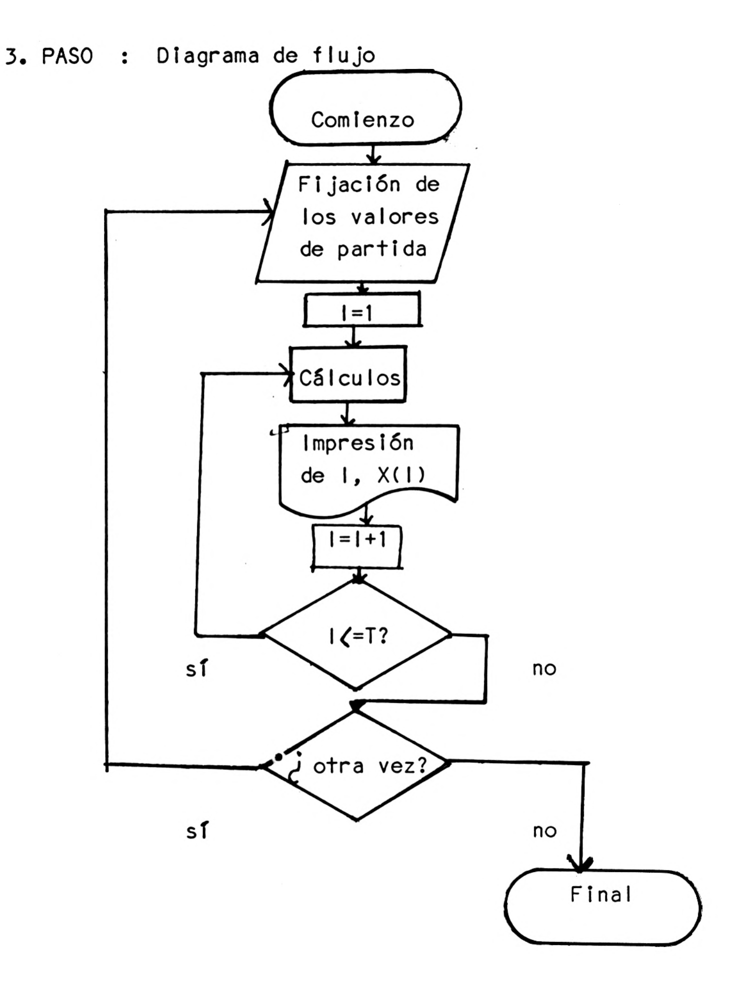
De este modo se cumple el siguiente algoritmo:

(1) B1 = B0 + A*X0

(2) X1 = X0 + X0 * (R/100 - P*B1)

Voss	Capítulo	6	:	Biología/Ecología	Página
	Punto	4	:	Contaminación	225
Colegio				ambiental	

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM B3-CONTAMINACION AMBIENTAL
20 CLS
30 PRINT "PROGRAMA PARA SIMULAR EN UNA TABLA"
40 PRINT TAB(2) "UN PROCESO DE CRECIMIENTO FRENADO"
50 PRINT TAB(2) "POR SU PROPIA DINAMICA"
60 PRINT: PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS, 1984"
70 PRINT: PRINT: PRINT: PRINT
80 PRINT "ESTE PROGRAMA REQUIERE LOS DATOS SIGUIENTES: ": PRINT
100 PRINT: INPUT "POBLACION INICIAL                    : "; X0
110 PRINT: INPUT "CUANTOS ANOS                         : "; T
120 PRINT: INPUT "TASA ANUAL DE CRECIMIENTO (%)        : "; R
120 PRINT: INPUT "FACTOR DE CONTAMINACION AMBIENTAL    : "; P
130 CLS
145 PRINT "TIEMPO  POBLACION  CRECIMIENTO (%)  CONTAMINACION":
    PRINT: PRINT
150 I=1: A=0.015: B0=0
160 B1 = B0+A*X0
170 X1 = X0 + X0 * (R/100 - P*B0)
175 IF X1 <= 0 THEN PRINT "POBLACION ELIMINADA": GOTO 210
180 Z = (X1-X0) / X0: Z = Z*100
190 PRINT TAB(2) I; TAB(7) USING "≠≠≠≠≠.≠#"; X1; :
    PRINT TAB(17) USING "≠≠≠≠.≠≠"; Z;
195 PRINT TAB(29) USING "≠≠≠≠≠"; B1
200 I=I+1: IF I <= T THEN B0=B1: X0=X1: GOTO 160
210 PRINT: PRINT: INPUT "OTRA VEZ (S/N)? "; A$
220 IF A$ = "S" THEN CLS: GOTO 80
230 PRINT: PRINT "FIN DE VISUALIZACION": END
```

5. PASO : Lista de variables

A = Factor contaminación ambiental depend ente de la población
A$ = Variable de cadena (si,no)
B0 = Contaminación ambiental de partida
B1 = Contaminación ambiental en el siguiente período
I = Indice variable
P = Factor contaminación ambiental
R = Tasa de crecimiento anual
T = Tiempo
X0 = Población de partida
X1 = Población en el siguiente período
Z = Crecimiento de la población

6. PASO : Descripción del programa

Línea 10-70 : Título y aclaraciones

Línea 80-120 : Demanda de las informaciones INPUT

Línea 130-145 : Borrado de la pantalla e impresión de los encabezamientos de la tabla

Línea 150 : Definición de otros parámetros

Línea 160-170 : Cálculos para el siguiente período

Línea 175 : Si la población calculada es menor

	o igual a cero, visualizar mensaje y bifurcar a la línea 210
Línea 180	Cálculo de Z
Línea 190 - 195	Visualización de resultados de los períodos
Línea 200	Incremento del índice variable I
	Si I es menor que T, X0 toma el valor de X1 y B0 el de B1, el programa en este caso retorna a la línea 160
Línea 210 - 230	Consulta si se desean más cálculos, (en caso afirmativo borrar la pantalla y volver a línea 80), y finalización del programa

7. PASO : Resultados

Sí a la población de partida le damos p.e. el valor 1000, al horizonte de tiempo el valor 8 años, a la tasa de crecimiento anual el 5% y como factor de contaminación ambiental tomamos el valor 0.01, obtendremos:

TIEMPO	POBLACION	CRECIMIENTO %	CONTAMINACION
1	1050	5	15
2	945	-10	30.75
3	701.66	-25.75	44.93
4	421.49	-39.93	55.45
5	208.85	-50.45	61.77
6	90.29	-56.77	64.9
7	36.21	-59.9	66.25
8	14.03	-61.25	66.79

OTRA VEZ (S/N) ?



Capítulo 7 : Geografía/Historia

7.1 Consideración previa

En este capítulo pueden confeccionarse programas similares a los del capítulo "lenguas", es decir, ante todo programas de test y de consulta. También aquí nos limitaremos a utilizar ficheros de datos de muestra, explicando el montaje y funcionamiento del programa.

Para su utilización práctica, habrá que crear primero ficheros de datos completos y después introducirlos. Con este fin se completarán los DATAS, modificando el valor de N según el caso.

7.2 Fechas históricas

1. PASO : Presentación del problema

Presentamos un programa que puede utilizarse para aprender fechas históricas y para repasar estas fechas.

Con este fin, el ordenador le presenta al usuario acontecimientos históricos, requiriéndole para que introduzca la fecha correcta de estos acontecimientos históricos. El usuario puede decidir cuántas preguntas le debe formular el ordenador por cada serie de tests.

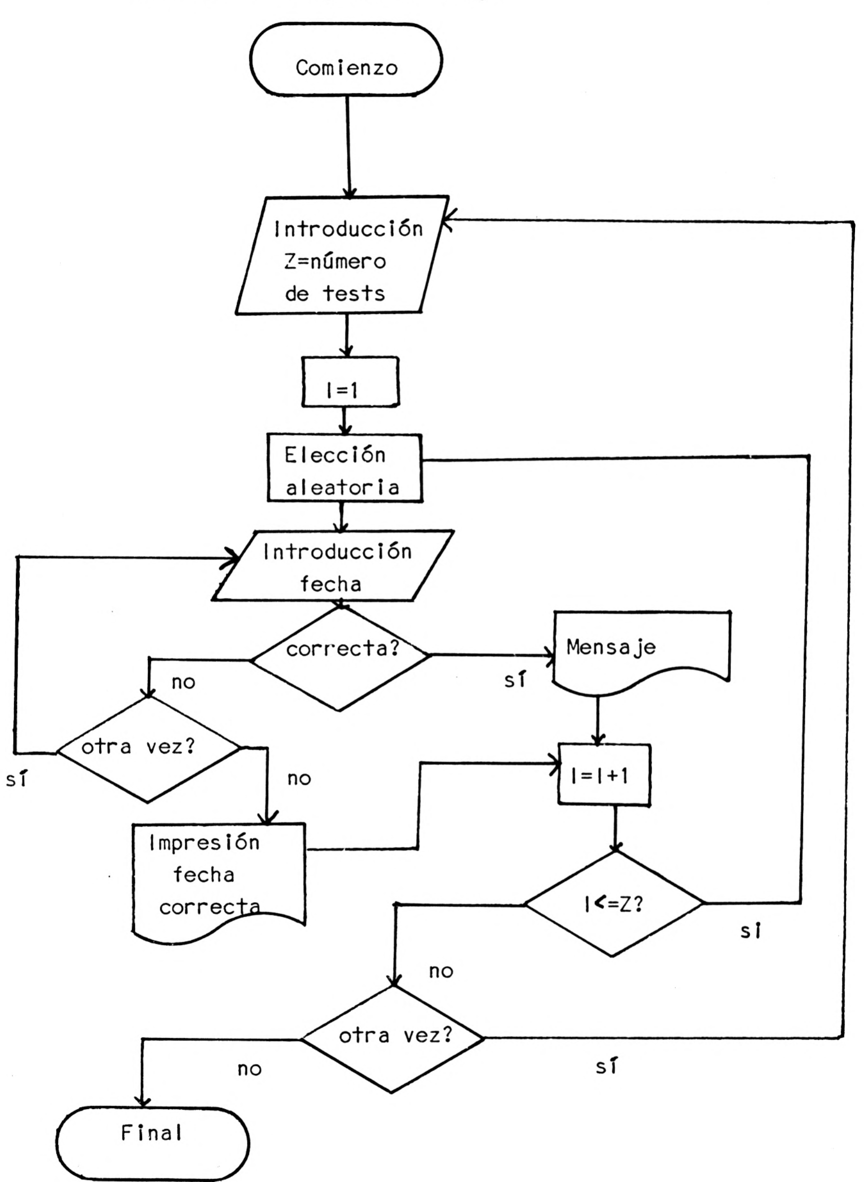
Si la respuesta del usuario es correcta, el ordenador le alaba; si, en cambio, la respuesta es incorrecta, el ordenador consulta si quiere probar otra vez. En caso de negativa por parte del usuario, el ordenador se ocupa de imprimir la respuesta correcta en pantalla.

2. PASO : Análisis del problema

La presentación del problema ya ha dado a entender claramente cómo debe procederse en este programa:

Una parte del programa está destinada a introducir información, también se debe preguntar al usuario cuántos tests desea hacer y el programa debe analizar cada respuesta. Por lo demás no se presentan dificultades especiales.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM E1-FECHAS HISTORICAS
20 CLS
30 PRINT "PROGRAMA PARA PEDIR DATOS HISTORICOS"
40 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS, 1984"
50 PRINT: PRINT: PRINT: PRINT
60 PRINT "ESTE PROGRAMA SE REFIERE UNICAMENTE A "
70 PRINT "10 DATOS HISTORICOS SELECCIONADOS"
80 PRINT "(VEANSE SENTENCIAS DATA EN 500 FF.)": PRINT
90  PRINT "EN CASO DE NECESITAR MAYOR CANTIDAD DE DATOS"
100 PRINT "AMPLIAR SENTENCIAS DATA A PARTIR DE LINEA 500"
110 PRINT "Y MODIFICAR VALOR DE N EN LINEA 150"
120 LOCATE 5,23
130 PRINT "PULSAR UNA TECLA POR FAVOR!"
140 A$=INKEY$: IF A$="" THEN 140
150 N=10
155 DIM J$(N),S$(N)
160 CLS
170 FOR I=1 TO N: READ J$(I),S$(I): NEXT I
180 PRINT "CUANTAS PREGUNTAS DESEA REALIZAR? "
190 PRINT: INPUT "POR FAVOR INDICAR NUMERO : "; Z
200 FOR I=1 TO Z
202 R=INT(RND(1)*N+1)
205 CLS
210 PRINT: PRINT "CUANDO FUE    : ": PRINT
230 PRINT S$(R);: PRINT
240 PRINT: PRINT "EN EL ANO     : ";: INPUT K$
250 IF K$=J$(R) THEN PRINT:  PRINT TAB(9) " P E R F.E C T O !  ":
    PRINT: GOTO 300
260 PRINT: PRINT "ES UNA LASTIMA, PERO NO ES CORRECTO! ": PRINT
270 PRINT: PRINT "UN INTENTO MAS ?";: INPUT " (S/N)? "; A$
280 IF A$ = "S" THEN 205
```

```
290 PRINT: PRINT "LA FECHA HISTORICA CORRECTA ES: "; J$(R)
300 FOR II = 1 TO 2000: NEXT II: NEXT I
310 PRINT: PRINT: PRINT "UN TEST MAS  ";: INPUT "(S/N)"; A$
320 IF A$ = "S" THEN CLS: GOTO 180
330 PRINT: PRINT "FINAL": END
500 DATA 1939,PRINCIPIO DE LA SEGUNDA GUERRA MUNDIAL
510 DATA 800,CORONACION DEL EMPERADOR CARLOMAGNO
520 DATA 1949,ESTABLECIMIENTO DE LA REPUBLICA FEDERAL
530 DATA 1808,GUERRA DE LA INDEPENDENCIA
540 DATA1492,DESCUBRIMIENTO DE AMERICA
550 DATA 1936,COMIENZO DE LA GUERRA CIVIL ESPANOLA
560 DATA 1871,ESTABLECIMIENTO DEL IMPERIO ALEMAN
570 DATA 1812,NAPOLEON ANTE MOSCU
580 DATA 1975,MUERTE DEL GENERALISIMO FRANCO
590 DATA 1914,PRINCIPIO DE LA PRIMERA GUERRA MUNDIAL
```

5. PASO : Lista de variables

A$	= Variable de cadena para respuestas
I	= Indice variable
II	= Indice variable
J$	= Fechas
N	= Número de sentencias DATA
R	= Número aleatorio
S$	= Hechos históricos
Z	= Número de tests

6. PASO : Descripción del programa

Línea 10 - 120	Títulos y aclaraciones
Línea 130 - 140	Pausa, espera
Línea 150 - 170	Preselección del número de sentencias DATA, dimensionamiento y borrado de la pantalla
Línea 180- 190	Requerimiento del número de tests

Línea 200 - 300 Tests:

Línea 202-240 : Elección al azar de un acontecimiento histórico y demanda de la fecha como respuesta del usuario.

Línea 250 : Si la respuesta es correcta, mensaje y continuación en 300.

Línea 260-280 : Si la respuesta no es correcta, mensaje y consulta si se desea otro intento (en caso afirmativo retorno a 205)

Línea 290 : En caso negativo, visualizar respuesta correcta.

Línea 300 : Bucle de espera 'y a continuación retorno a línea 200.

Línea 310 - 330 Consulta si desea otra serie de tests; en caso afirmativo, retorno a línea 180 después de borrar la pantalla, en caso negativo, final del programa.

Línea 500 - 590 Datos

7. PASO : Resultados

No es necesario ofrecer aquí los resultados. Nos parece más razonable que el lector pruebe el programa considerando los contenidos y las DATAS, modificándolo después según sus propios deseos.

7.3 Las capitales de las Comunidades Autónomas

1. PASO : Presentación del problema

El programa que presentamos en este punto es también un llamado programa de entreno. El usuario debe responder a las preguntas que le formula el ordenador. Si el ordenador ofrece nombres de comunidades autónomas elegidos al azar, el usuario deberá introducir las capitales de estas comunidades; si en cambio el ordenador visualiza capitales, el usuario deberá citar las correspondientes comunidades.

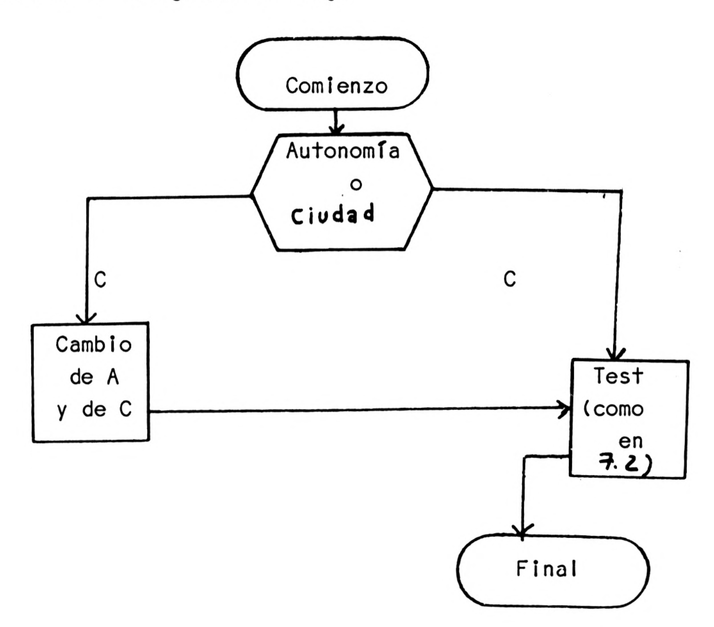
Llegados a este punto no hay problemas nuevos, de modo que podremos ser breves. Este programa tiene carácter ejemplar :

El usuario puede registrar fácilmente nuevos datos en los DATAS, obteniendo así un programa de consulta de aplicación general, que puede utilizarse para las más diversas asignaturas escolares.

2. PASO : Análisis del problema

Ya hemos mencionado que aquí no se presenta ningún problema adicional, de modo que podemos pasar rápidamente al siguiente paso. Todo lo referente a este programa ya ha sido suficientemente discutido en ejemplos precedentes.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM G2-COMUNIDADES AUTONOMAS
20 CLS
30 PRINT "  CONTROL  DE  GEOGRAFIA  "
40 PRINT: PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS, 1984"
60 PRINT: PRINT: PRINT: PRINT
70 PRINT "ESTE PROGRAMA TRAS INDICAR UNA COMUNIDAD AUTONOMA"
80 PRINT "NOS PREGUNTA SU CAPITAL Y VICEVERSA.": PRINT: PRINT
90 PRINT "ESTE PROGRAMA PUEDE UTILIZARSE "
100 PRINT "TAMBIEN PARA OTROS CONTROLES "
110 PRINT "SI SE MODIFICAN LAS SENTENCIAS DATA"
120 PRINT "A PARTIR DE LA LINEA 500 "
130 PRINT "Y EL VALOR DEL CONTADOR N EN LINEA 160"
140 LOCATE 5,23: PRINT "PULSAR UNA TECLA POR FAVOR"
150 A$=INKEY$: IF A$="" THEN 150
160 N=7
170 DIM L$(N),S$(N)
180 FOR I=1 TO N: READ L$(I),S$(I):NEXT I
190 CLS
200 PRINT "DEBEN INDICARSE AUTONOMIAS (A) O CAPITALES (C)"
210 INPUT "ENTRAR A/C"; A$
215 C$="CAPITAL"
220 IF A$="C" THEN C$ = "AUTONOMIA": GOTO 250
230 GOSUB 1000: REM TEST
240 GOTO 270
250 FOR I=1 TO N: H$=L$(I): L$(I)=S$(I): S$(I)=H$: NEXT I
260 GOSUB 1000: REM TEST
270 PRINT: PRINT: INPUT "OTRA VEZ (S/N): "; A$
280 IF A$="S" THEN CLS: RESTORE: GOTO 180
290 PRINT: PRINT "FINAL": END
500 DATA CATALUNYA,BARCELONA,MADRID,MADRID
510 DATA RIOJA,LOGRONO,BALEARES,PALMA
```

```
520 DATA P. VALENCIANO,VALENCIA,EUZCADI,GASTEIZ
530 DATA CANTABRIA,SANTANDER,ANDALUCIA,SEVILLA
540 DATA GALICIA, LA CORUNA,ASTURIAS,GIJON
550 DATA EXTREMADURA,?,CANARIAS,LAS PALMAS
560 DATA MURCIA,MURCIA,CASTILLA-LAMANCHA,??
570 DATA CASTILLA-LEON,LEON,ARAGON,ZARAGOZA
1000 REM UP TEST
1010 CLS
1020 PRINT: INPUT "CUANTAS PREGUNTAS : "; Z
1030 FOR I=1 TO Z
1040 R=INT(RND(1)*N+1)
1050 PRINT: PRINT L$(R): PRINT: PRINT: PRINT C$; " : ";: PRINT
     : PRINT: INPUT X$
1060 IF X$=S$(R) THEN PRINT: PRINT TAB(10) "C O R R E C T O":
     PRINT :GOTO 1100
1070 PRINT: PRINT TAB(10) "ERROR! ": PRINT
1080 PRINT: PRINT "LA RESPUESTA CORRECTA ES : "
1090 PRINT: PRINT TAB(10) S$(R): PRINT
1100 NEXT I
1110 RETURN
```

5. PASO : Lista de variables

A$ = Variable de cadena para respuestas
C$ = Variable auxiliar que según opción, puede tomar el valor "capital" o "autonomía"
I = Indice variable
L$ = Autonomías
N = Número de registros
S$ = Capitales

6. PASO : Descripción del programa

Línea 10 - 130 Título y aclaraciones

Línea 140 - 150 Espera

Línea 160 - 180 Lectura de las sentencias de datos

Línea 190 - 210 Consulta si deben visualizarse autonomías o capitales

Línea 220 Si se seleccionan 'capitales', la variable C$ pasa a ser 'autonomía' y tiene lugar una bifurcación a línea 250

Línea 230 Bifurcación a subrutina de línea 1000 donde se efectúa el test.

Línea 240 Bifurcación a línea 270

Línea 250	Intercambio entre 'capitales' y 'autonomías' para utilizar el mismo programa en ambos sentidos.
Línea 260	Como línea 230
Línea 270 - 290	Consulta si se quiere reanudar la serie de test, (en caso afirmativo se borra la pantalla y se regresa a línea 180 una vez restaurados índices y tablas); en caso contrario termina el programa.
Línea 500 - 570	Datos
Línea 1000 - 1110	Subrutina test: Programa para realizar los test, según las opciones entradas.

7. PASO : Resultados

Tampoco en este programa resulta necesario ofrecer los resultados. El análisis del programa, o aún mejor, la ejecución del mismo muestra claramente lo que ocurre en cada momento.

7.4 El desarrollo de la población en diferentes países

1. PASO : Presentación del problema

Presentamos aquí un programa que pronostica el desarrollo de la población de diferentes países del mundo hasta el año 2000.

Los países o grupos de países seleccionados para el estudio son:

Mundo (comprendiendo a todos los países)

Países industrializados
Países en vías de desarrollo

RP China
India
Japón
USA
URSS

República Federal de Alemania
RDA.

2. PASO : Análisis del problema

Los pronósticos de población planteados réquieren el decidirse por un determinado método de pronosticación.

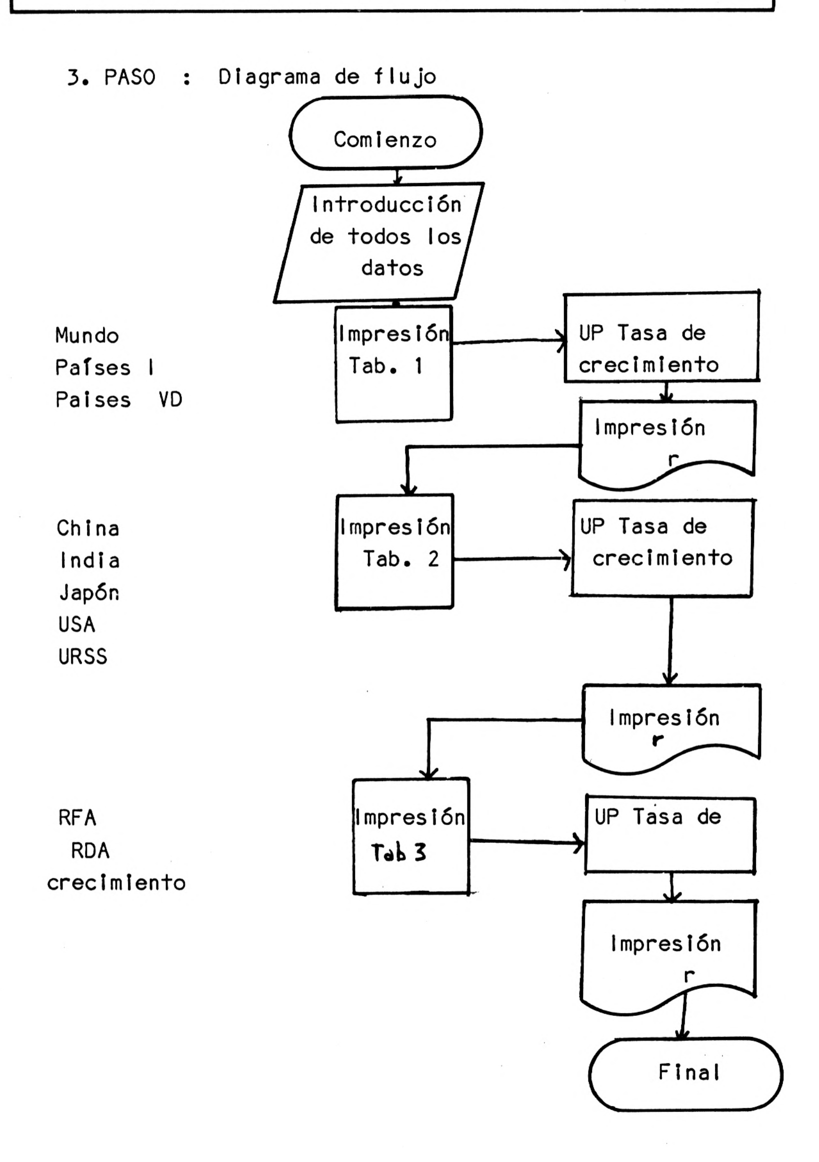
Existe un gran número de métodos distintos, que aquí no pueden ser discutidos uno a uno.

Para este programa hemos extraído los valores obtenidos en pronósticos modelo como los publicados en el informe GLOBAL 2000 (Washington 1980).

A partir de estos datos calculamos además las tasas de crecimiento medias anuales, para poder ofrecer a través del programa una información adicional.

Desde el punto de vista de la programación no aparecen dificultades importantes, si exceptuamos que, debido al volumen de resultados, en diversos puntos del programa es necesario efectuar interrupciones.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM   G3-POBLACION
20 CLS
30 PRINT "PROGRAMA PARA REPRESENTAR EN UNA"
40 PRINT "TABLA EL DESARROLLO DE LA POBLACION"
50 PRINT "DE DIFERENTES PAISES"
60 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS, 1984"
70 DIM B(10,6),J(6),L$(10),WR(10)
80 FOR I = 1 TO 6:  READ J(I): NEXT I
90 FOR L = 1 TO 10: FOR I = 1 TO 6: READ B(L,I): NEXT I: NEXT L
100 FOR L = 1 TO 10: READ L$(L): NEXT L
110 GOSUB 1000: REM ESPERA
140 PRINT "ANO     MUNDO      IND.    DES."
145 PRINT
150 GOSUB 1200: REM RAYA
160 PRINT
170 FOR I =  1 TO 6
180 PRINT J(I);TAB(8)B(1,I);TAB(18)B(2,I);TAB(32)B(3,I): PRINT
190 NEXT I
195 GOSUB 1200 : REM RAYA
200 FOR L = 1 TO 3
210 X=B(L,1): Y=B(L,6)
220 GOSUB 1100: REM TASA CRECIMIENTO
230 NEXT L
240 PRINT: PRINT  "TASA  %"; : PRINT TAB(8) WR(1); TAB(18) WR(2);
    TAB(32) WR(3): PRINT
250 PRINT: GOSUB 1000: REM ESPERA
260 CLS
270 PRINT "ANO      CHINA    INDIA    JAPON    USA    URSS": PRINT
280 GOSUB 1200: REM RAYA
290 FOR I = 1 TO 6
300 PRINT J(I) ;TAB(8)B(4,I);TAB(14)B(5,I);TAB(21)B(6,I);
302 PRINT TAB(28) B(7,I); TAB(34) B(8,I): PRINT
```

```
310 NEXT I
320 GOSUB 1200: REM RAYA
330 FOR L = 4 TO 8
340 X = B(L,1): Y = B(L,6)
350 GOSUB 1100: REM TASA DE CRECIMIENTO
360 NEXT L
370 PRINT: PRINT "%";
380  PRINT TAB(8) WR(4);  TAB(14) WR(5);  TAB(21)  WR(6);
     TAB(28) WR(7); TAB(34) WR(8)
385 PRINT: PRINT
390 GOSUB 1000: REM PAUSA
400 CLS
410 PRINT "ANO    RFA   DDR": PRINT
420 GOSUB 1200: REM RAYA
430 FOR I=1 TO 6
440 PRINT J(I); TAB(12) B(9,I); TAB(24) B(10,I): PRINT
450 NEXT I
460 GOSUB 1200: REM RAYA
470 FOR L = 9 TO 10
480 X = B(L,1): Y = B(L,6)
490 GOSUB 1100: REM TASA DE CRECIMIENTO
492 NEXT L
494 PRINT "%";: PRINT TAB(12) WR(9); TAB(24) WR(10)
496 PRINT: PRINT "FINAL": END
500 DATA 1975,1980,1985,1990,1995,2000
505 REM MUNDO
510 DATA 4134,4549,5013,5545,6143,6798
515 REM PAISES INDUSTRIALIZADOS
520 DATA 1131,1174,1224,1276,1327,1377
525 REM PAISES NO INDUSTRIALIZADOS
530 DATA 3003,3375,3789,4269,4816,5420
535 REM RP CHINA
540 DATA 978,1071,1151,1241,1348, 1468
545 REM INDIA
550 DATA 618,694,786,894,1013,1142
```

```
555 REM JAPON
560 DATA 112,117,122,127,131,135
565 REM USA
570 DATA 214,222,235,248,260,270
575 REM URSS
580 DATA 254,268,282,296,310,323
585 REM RFA
590 DATA 61.8,61.7,60.0,58.6,57.8,56.2
595 REM RDA
600 DATA 16.8,16.7,16.6,16.4,16.2,16.1
610 DATA "MUNDO","IND.","DES.","CHINA","INDIA"
620 DATA "JAPON","USA","URSS","RFA","RDA"
1000 REM ESPERA
1010 LOCATE 5,23: PRINT "PULSE UNA TECLA POR FAVOR"
1020 A$=INKEY$: IF A$="" THEN 1020
1030 CLS
1040 RETURN
1100 REM TASA DE CRECIMIENTO
1110 R=EXP(LOG(Y/X)/25)-1
1120 R=R*100: R=INT(R*100+0.5)/100
1130 WR(L)=R
1140 RETURN
1200 REM RAYA
1210 FOR I=1 TO 40: PRINT "-";: NEXT I
1220 PRINT: RETURN
```

5. PASO : Lista de variables

B = Población
I = Indice variable
J = Año
L = Indice variable
L$ = Países

WR = Tasa de crecimiento
X = Valor de partida (1975)
Y = Valor final (2000)

6. PASO : Descripción del programa

Línea 10 - 60	Títulos
Línea 70 - 100	Lectura de todos los datos de partida
Línea 110	Bifurcación a sentencia 1000 y pausa
Línea 140 - 145	Encabezamiento de la tabla
Línea 150	Bifurcación a sentencia 1200 , subrayado
Línea 160 - 190	Visualización de los valores de la tabla
Línea 200 - 230	Determinación de las tasas de crecimiento medias anuales para los países L=1, L=2 y L=3 Línea 210 : Toma los valores inicial y final Línea 220 : Bifurca a línea 1100; donde calcula la tasa de crecimiento
Línea 240	Visualización de tasas de crecimiento
Línea 250	Espera
Línea 260 - 390	Mismo procedimiento que en línea 140-250 pero para la segunda tabla
Línea 400 - 494	Igual procedimiento para la tercera tabla
Línea 496	Final del programa
Línea 500 - 620	Datos
Línea 1000 - 1040	Subrutina de espera
Línea 1100 - 1140	Subrutina tasa de crecimiento medio anual (se calcula por medio de logaritmos, ver línea 1110)
Línea 1200 - 1220	Subrutina de subrayado: Mediante encadenamiento de signos "-" se dibuja una raya en la pantalla.

7. PASO : Resultados

Este programa genera en la pantalla tres tablas consecutivas :

1. Tabla (referencias en millones)

AÑO	MUNDO	PAISES I	PAISES VD
1975	4134	1131	3003
1980	4549	1174	3375
1985	5013	1224	3789
1990	5545	1276	4269
1995	6143	1327	4816
2000	6798	1377	5420
TASA %	2.01	.79	2.39

2. Tabla (referencias en millones)

AÑO	CHINA	INDIA	JAPON	USA	USSR
1975	978	618	112	214	254
1980	1071	694	117	222	268
1985	1151	786	122	235	282
1990	1241	894	127	248	296
1995	1348	1013	131	260	310
2000	1468	1142	135	270	323
%	1.64	2.49	.75	.93	.97

3. Tabla (referencias en millones)

AÑO	RFA	RDA
1975	61.8	16.8
1980	61.7	16.7
1985	60	16.6
1990	58.6	16.4
1995	57.8	16.2
2000	56.2	16.1
%	-.38	-.17

FINAL

8

Capítulo 8 : La economía

8.1 Consideración previa

Durante mucho tiempo, el campo de aplicación fundamental de los ordenadores fue el de la Economía, y en especial el empresarial y comercial. Puede decirse que hoy en día sigue siendo en estos campos, donde se emplean con más frecuencia los ordenadores.

Los problemas tratados en estos campos se prestan a ser resueltos por medio de ordenadores, hoy en día también con microordenadores.

Algunos de los problemas fundamentales de la Economía, por lo menos en lo referente a los cálculos, son tratados también en las escuelas; en escuelas empresariales de forma más intensiva, pero también en otros tipos de escuelas, donde tampoco se evitan este tipo de problemas.

También aquí, los problemas seleccionados sólo son representativos de otros muchos, que por motivos de espacio no pueden ser incluídos en esta obra.

8.2 El cálculo de intereses

1. PASO : Presentación del problema

Imaginemos que alguien lleva un determinado importe al banco, del que se deducirá el pago de un determinado interés por la imposición del mismo. Este interés puede ser fijado arbitrariamente.

Al final de cada año aumenta el capital en un determinado tanto por ciento. Hay que tener en cuenta, además, que a partir del segundo año se obtiene un interés sobre el capital más el interés del año anterior (interés compuesto).

Vamos a desarrollar ahora un programa que calculará - para un importe inicial arbitrario y para un rédito cualquiera- el importe final obtenido después de una cifra determinada de años.

Una programa como este puede servir además como patrón para la simulación de diversos procesos de crecimiento.

2. PASO : Análisis del problema

Si llamamos al capital de partida K0, al rédito P (%) y al tiempo que se tiene el dinero colocado en el banco T (años).

Después del primer año, el capital resultante X1 es :

$$X1 = X0 + X0 * P/100 = X0 * (1 + P/100)$$

Tras el segundo año se obtiene del mismo modo:

$$X2 = X1 + X1 * P/100 = X1 * (1 + P/100)$$

$$= X0 * (1 + P/100) * (1 + P/100)$$

$$= X0 * (1 + P/100)^2$$

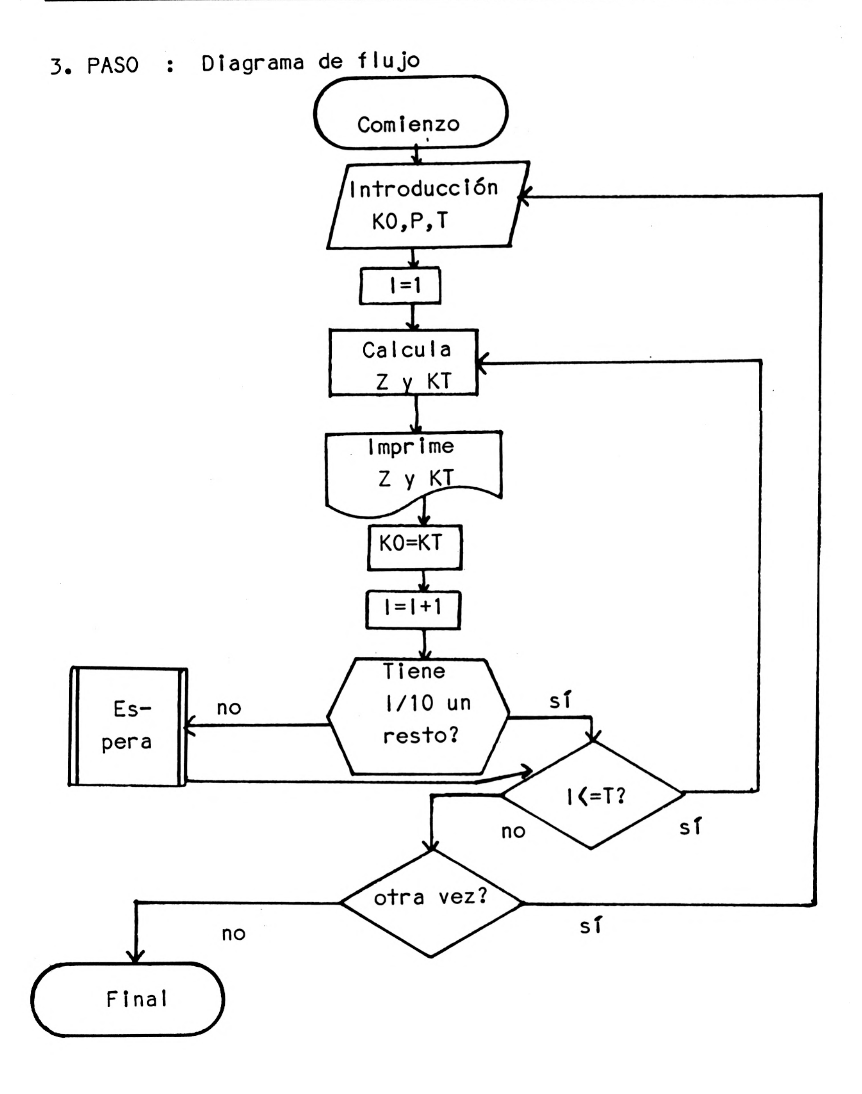
(Estas relaciones ya las conocemos de capítulos anteriores).

De esta forma, después de T años se obtiene:

$$X_T = X0 * (1 + P/100)^T$$

Con ayuda de esta fórmula podemos confeccionar ahora un programa adecuado.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM E1-CALCULO DE INTERESES
20 CLS
30 PRINT "PROGRAMA PARA CALCULO DE INTERESES"
40 PRINT "CALCULA EL INTERES ANUAL"
50 PRINT "INTRODUCIENDO EL CAPITAL CORRESPONDIENTE"
60 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS, 1984"
70 PRINT: PRINT: PRINT: PRINT: PRINT
80 INPUT          "CAPITAL INICIAL       : "; K0
90 PRINT:  INPUT "REDITO ANUAL EN %     : "; P
100 PRINT: INPUT "CANTIDAD DE ANOS      : "; T
110 CLS
120 PRINT "ANO     INTERESES      CAPITAL": PRINT: PRINT
130 FOR I=1 TO T
150 Z = K0*P/100
160 KT = K0 + Z
190 PRINT TAB(2) I;:PRINT TAB(8) USING "≠≠≠≠.≠≠"; Z;:
    PRINT TAB(18) USING "≠≠≠≠≠≠.≠≠"; KT
200 K0 = KT
210 IF I/10=INT(I/10) THEN GOSUB 500: REM ESPERA
220 NEXT I
230 PRINT: PRINT: PRINT "OTRO CALCULO? (S/N)";; INPUT A$
240 IF A$="S" THEN CLS: GOTO 80
250 PRINT: PRINT: PRINT "FINAL DEL CALCULO": END
500 REM UP ESPERA
510 LOCATE 5,23
520 PRINT "PULSE UNA TECLA, POR FAVOR!"
530 A$=INKEY$: IF A$="" THEN 530
540 CLS: PRINT "ANO    INTERESES    CAPITAL": PRINT: PRINT
550 RETURN
```

5. PASO : Lista de variables

A$ = Variable de cadena para las respuestas
I = Indice variable
KO = Capital inicial
KT = Capital al final de cada período
P = Rédito anual en porcentaje
T = Tiempo de imposición en anos
Z = Interés anual

6. PASO : Descripción del programa

Línea 10 - 70 Título

Línea 80 - 100 Demanda de los datos de partida

Línea 110 - 120 Limpieza de la pantalla y visualización de un encabezamiento de tabla

Línea 130 - 220 Cálculos

150 : Interés
160 : Capital tras el siguiente período
190 : Salida a pantalla
200 : Ko toma el valor KT, para poder entrar en el siguiente cálculo
210 : Interrupción mediante bifurcación a 500 después de cada 10 períodos

Línea 230-250 : Final del programa, a menos que se desee seguir calculando (vuelta a la línea 80)

Línea 500-550 : Subrutina para esperar cuando la pantalla está llena:

510-520	:	Indicación
530	:	Espera
540-546	:	Borrado de la pantalla y nueva impresión del encabezamiento
550	:	Salto de vuelta

7. PASO : Resultados

Si introducimos el capital inicial Ptas. 100, el rédito 5.5% y el tiempo 6 años, se obtiene :

AÑO	INTERES	CAPITAL
1	5.5	105.5
2	5.2	113.3
3	6.12	117.42
4	6.46	123.88
5	6.81	130.69
6	7.19	137.88

MAS CALCULOS (S/N) ?

8.3 Amortización de una hipoteca

1. PASO : Presentación del problema

De entre los problemas de la Economía, aquéllos que están relacionados con el pago de créditos e hipotecas juegan también un papel importante. Hay programas de ordenador que pueden calcular el tiempo de amortización bajo diversas condiciones de partida, como p.e. valor de la deuda, el interés o las modalidades de pago.

El programa siguiente calcula el tiempo de amortización para diferentes importes e intereses o cuando se han acordado cuotas de amortización de distintas cuantías.

Puede caerse fácilmente en la cuenta de que este programa es aplicable de forma más general en problemas de este tipo.

2. PASO : Análisis del problema

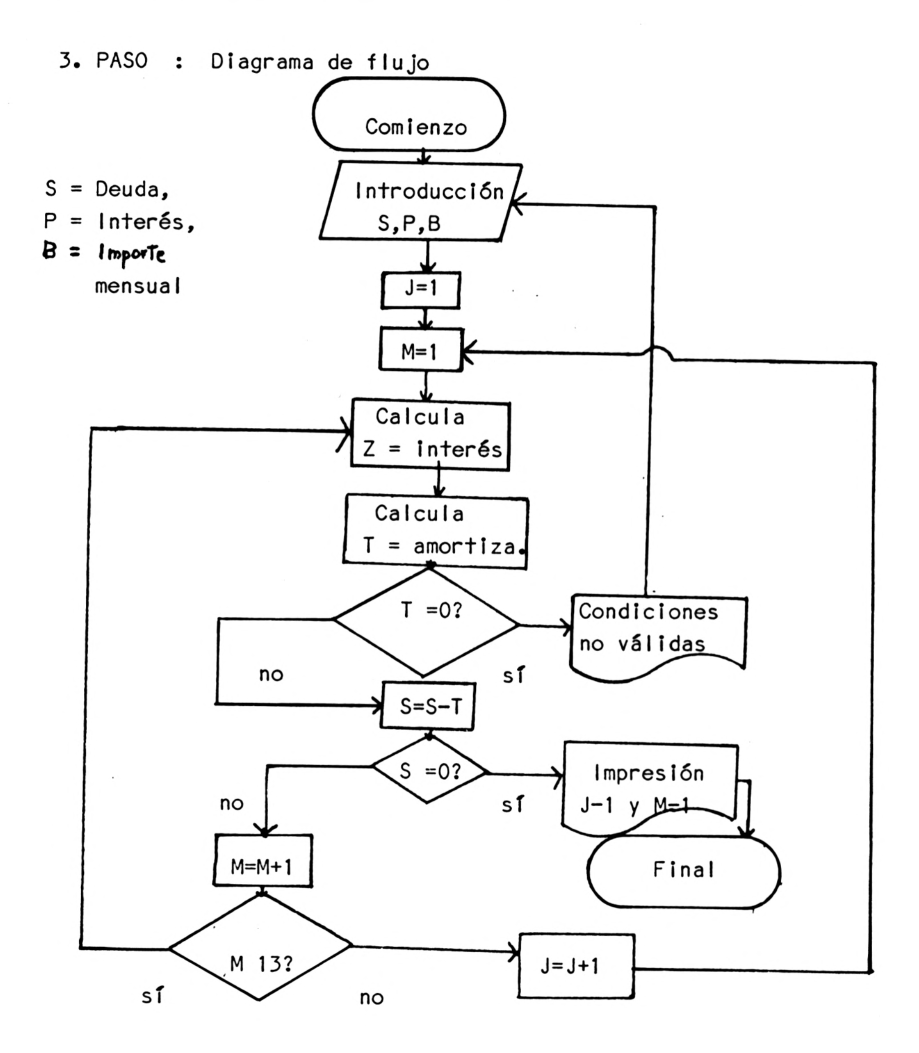
Para explicar la resolución del problema nos basaremos en el siguiente ejemplo:

Alguien solicita una hipoteca de S Ptas. y acuerda con el banco concesor un pago mensual de B Ptas. De este importe mensual B deben restarse primero los intereses por la deuda. El importe que queda después de restar los intereses puede aplicarse para saldar la deuda - suponiendo que quede un resto positivo.

El programa que vamos a confeccionar debe calcular mes a mes los intereses y el capital de amortización. Si este importe es igual o menor a 0, el programa debe imprimir un mensaje para avisar de que bajo esas condiciones ya no es posible amortizar la deuda.

Debe imprimirse año por año un estado de la deuda, y cuando la deuda sea cero o negativa, deberá indicarse la duración del proceso de amortización.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM E2 - HIPOTECA
20 CLS
30 PRINT "PROGRAMA PARA CALCULAR EL TIEMPO"
40 PRINT "DE AMORTIZACION DE UNA HIPOTECA"
50 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS,1984"
60 PRINT: PRINT: PRINT: PRINT
70 PRINT: INPUT "CUOTA MENSUAL              : "; B
80 PRINT: INPUT "INTERES EN %               : "; P
90 PRINT: INPUT "DEUDA HIPOTECARIA ACTUAL : "; S
100 CLS
110 PRINT "ANO      DEUDA": PRINT
120 FOR I = 1 TO 20: PRINT "-";: NEXT I: PRINT
130 J = 1
140 M = 1
150 Z = ((P/100)*S/12: T = B-Z
160 IF T <= 0 THEN 250
170 S = INT((S-T)*100+0.5)/100
180 IF S <= 0 THEN 300
190 M = M+1: IF M < 13 THEN 150
200 PRINT TAB (2) J; TAB(8) S
210 J = J + 1: GOTO 140
250 PRINT: PRINT: PRINT "LAS CONDICIONES NO PERMITEN "
260 PRINT "AMORTIZAR LA DEUDA"
290 PRINT: PRINT "NUEVOS DATOS POR FAVOS: PRINT: GOTO 70
300 PRINT: PRINT: PRINT "TIEMPO DE AMORTIZACION : ": PRINT
310 PRINT J - 1; "ANO Y "; M - 1; " MES."
320 PRINT: PRINT: PRINT "OTRO CALCULO? (S/N) ";: INPUT A$
330 IF A$="S" THEN CLS: GOTO 70
340 PRINT: PRINT: PRINT "FIN DEL CALCULO": END
```

5. PASO : Lista de variables

A$ = Variable de cadena (sí,no)
B = Cuota mensual
I = Indice variable
J = Contador de años
M = Contador de meses
P = Interés anual de la deuda
T = Importe mensual de amortización
Z = Pago mensual de intereses

6. PASO : Descripción del programa

Línea 10-60 : Título

Línea 70-90 : Visualización de las informaciones de partida

Línea 100-120 : Impresión del encabezamiento de la tabla

Línea 130-140 : Fijación del contador de años (J) y del contador de meses M

Línea 150-170 : Cálculo del nterés mensual, de la cuota mensual y de la nueva deuda

Línea 160 : En caso de no poder amortizar, seguir en 250

Línea 180 : Una vez saldada la deuda, seguir en 300

Línea 190 : Mes siguiente
si N es menor que 13 seguir en 150, en caso contrario seguir en la línea 200

Línea 200 : Impresión de la deuda actual

Línea 210 : Año siguiente y seguir en línea 140

Línea 250-290 : Mensaje sobre condiciones no permitidas y vuelta a la línea 70

Línea 300-310 : Impresión del tiempo de amortización en años y en meses (piensa porque se le resta cada vez el valor 1 a J y a N)

Línea 320-340 : Final del programa, a menos que no se desee un nuevo cálculo (entonces retor-

7. PASO : Resultados

Introduzcamos estos datos cuando el ordenador nos lo pida:

```
CUOTA MENSUAL       :  ? 500
INTERES EN %        :  ? 6.5
DEUDA HIP ACTUAL    :  ? 40000
```

y obtendremos en el transcurso del programa :

```
AÑO     DEUDA
--------------------------
 1       36496.86
 2       32759.09
 3       28771.02
 4       24515.85
 5       19975.7
 6       15131.48
 7       9962.85
 8       4448.07

TIEMPO DE AMORTIZACION :

 8 AÑOS Y 9 MESES.

OTRO CALCULO  (S/N) ?
```

8.4 Media aritmética

1. PASO : Presentación del problema

En algunos problemas de economía resulta necesario calcular el promedio de un número determinado de datos; normalmente se trata de la media aritmética.

El programa siguiente efectúa este cálculo para un número cualquiera de datos.

2. PASO : Análisis del problema

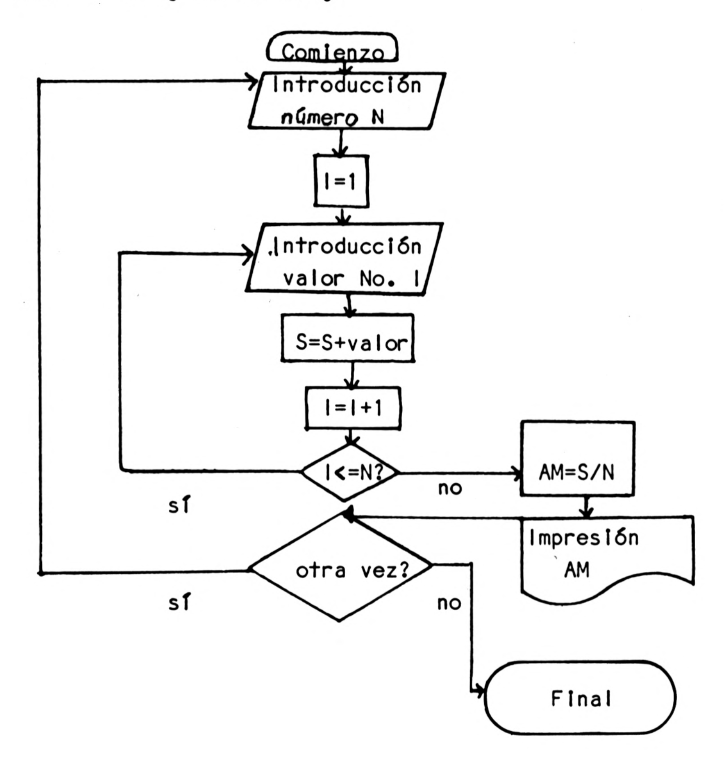
El análisis del problema no presenta especiales dificultades.

Primero hay que procurar que se introduzca un número no fijado de datos, por lo que se ha de informar al ordenador de cuántos valores se van a introducir.

Al introducir los valores puede efectuarse automáticamente la suma acumulativa de los mismos (la media aritmética no es más que la suma de todos los valores dividida entre el número de ellos). La división de la suma resultante del número de valores puede imprimirse como resultado del programa.

Para poder efectuar más cálculos, el programa se bifurca al principio. Si no se desea continuar finaliza él mismo.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM E3 - VALOR MEDIO
20 CLS
30 PRINT "PROGRAMA PARA CALCULAR UNA "
40 PRINT "MEDIA ARITMETICA "
50 PRINT: PRINT: PRINT TAB (9) "PROF.DR.W. VOSS, 1984"
60 PRINT: PRINT: PRINT: PRINT: PRINT
70 INPUT "CUANTOS VALORES : "; N
80 CLS: S=0
90 FOR I=1 TO N
100 PRINT I; " . VALOR  : ";: INPUT X
110 S = S + X
120 NEXT I
130 AM = S/N
150 PRINT: PRINT: PRINT "MEDIA = "; AM: PRINT
160 PRINT: PRINT: PRINT "OTRO CALCULO? (S/N) ";: INPUT A$
170 IF A$ = "S" THEN CLS: GOTO 70
180 PRINT: PRINT: PRINT "FINAL DEL CALCULO":END
```

5. PASO : Lista de variables

AM = Media aritmética
A$ = Variable de cadena (si/no)
I = Indice variable
N = Número de valores a promediar
S = Suma de los valores a promediar
X = Valor que entra en el cálculo del valor medio

6. PASO : Descripción del programa

Línea 10-60 : Título

Línea 70 : Indicación del número de valores a promediar

Línea 80 : Borrado de la pantalla y visualización del campo de suma

Línea 90-120 : Introducción de un valor tras otro y suma acumulativa de todos los valores

Línea 130-150 : Cálculo de la media aritmética e impresión del resultado

Línea 160-180 : Final del programa, a menos que no se desee continuar calculando

7. PASO : Resultados

Si introducimos p.e., que existen tres valores, el programa pedirá exactamente tres valores.

Si éstos son p.e. los valores 5, 7 y 9, obtendremos:

VALOR MEDIO = 7

OTRO CALCULO (S/N) ?

8.5 Distribución de frecuencias

1. PASO : Presentación del problema

A la hora de valorar volúmenes de datos grandes, que es cuando los ordenadores se muestran especialmente útiles, uno de los primeros pasos de la investigación estadística consiste en realizar una llamada distribución de frecuencias.

Se trata de asignar mediante una tabla o un gráfico la frecuencia con que aparecen los valores característicos (o a las clases de valores característicos) de la variable objeto de la investigación.

El programa siguiente realiza esta distribución de frecuencias tanto gráficamente como en forma de tabla. Como material de valoración utilizaremos la distribución por edades de la población de la RFA en el año 1980.

Esta distribución por edades de la población federal alemana es la siguiente:

Distribución de edades en la República Federal de Alemania en 1980:

Grupos de edades	Porcentaje
menos de 10	10.3
10 hasta 20	16.4
20 hasta 30	14.6
30 hasta 40	13.4
40 hasta 50	14.2
50 hasta 60	11.9
60 hasta 70	8.9
70 hasta 80	7.8
80 hasta 90	2.4
90 y más	0.2

2. PASO : Análisis del problema

Para el análisis del problema partiremos de la base de que disponemos de los datos tal como aparecen en la tabla, de forma que puedan ser utilizados como material de partida para el programa a confeccionar.

De no ser este el caso, y tuviéramos datos sueltos obtenidos en una encuesta, deberíamos procurar en una primera parte del programa la clasificación de estos datos sueltos por grupos de edades (esto puede conseguirse mediante una serie de instrucciones IF...THEN, que comprobarían cada uno de los valores de edad).

Una vez hecho esto, podemos relativizar los valores obtenidos para cada grupo de edades, comparándolos con la totalidad. De este modo, obtenemos los porcentajes que ya aparecen en la tabla anterior.

Esta parte del programa podría resumirse esquemáticamente de la forma siguiente:

```
100 INPUT "Edad siguiente : ";X
110 IF X<10 THEN Z1=Z1+1 : GOTO 100
120 IF X<20 THEN Z2=Z2+1 : GOTO 100
...
etc.
200 R1=(Z1/N)*100
210 R2=(Z2/N)*100
...
etc.
```

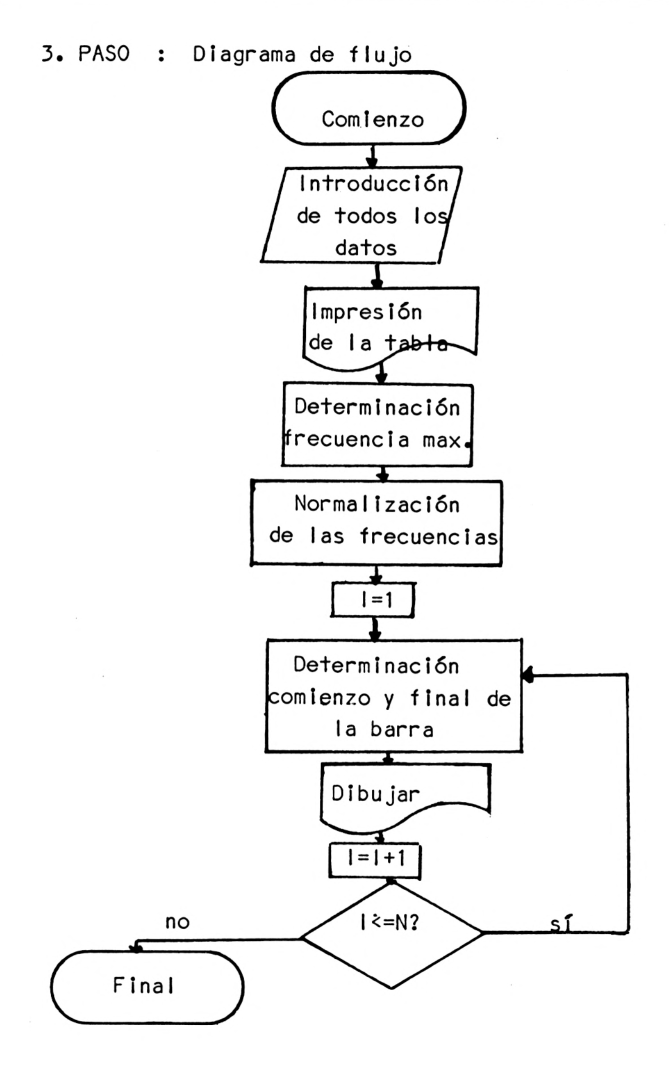
En este ejemplo se trata únicamente de representar en forma de tabla los valores introducidos, cosa que lógicamente no ofrece ninguna dificultad desde el o de vista de la programación.

La representación gráfica de esta distribución de frecuencias, en cambio, es más complicada.

Para poder aprovechar óptimamente la pantalla en la representación gráfica, determinaremos, en primer lugar,cuál es la frecuencia mayor y, a continuación, convertiremos todas las demás frecuencias de forma que la primera ocupe 15 líneas en vertical de la pantalla.

Las frecuencias en sí se representan en forma de barras verticales. Para que el tamaño de estas barras sea correcto, debe determinarse su correspondiente punto final. Junto con el punto de origen obtendremos dos posiciones de pantalla por barra y ya tan solo será necesario conectar con una línea.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM E4 - DISTRIBUCION
20 CLS
30 PRINT "PROGRAMA PARA REPRESENTAR EN UNA TABLA Y "
40 PRINT "GRAFICAMENTE UNA DISTRIBUCION ESTADISTICA"
50 PRINT "DE FRECUENCIAS"
60 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS, 1984"
70 PRINT: PRINT: PRINT: PRINT: PRINT
80 PRINT "SE REPRESENTA LA DISTRIBUCION DE EDADES EN LA"
90 PRINT "REPUBLICA FEDERAL ALEMANA EN 1980"
110 GOSUB 1000 : REM ESPERA
120 DIM F(10), FT(10)
130 FOR I = 1 TO 10: READ F(I): NEXT I
140 PRINT "EDAD"; TAB(22) "PORCENATAJE EN % ": PRINT
150 FOR I = 1 TO 35: PRINT "-";: NEXT I: PRINT
160 FOR I = 1 TO 10
165 A = (I-1)*10: B = I*10
170 PRINT USING "##"; A;
172 PRINT TAB(6) " HASTA MENOS DE ";
174 PRINT USING "###"; B;
176 PRINT TAB(24) USING "###.#" F(I)
180 NEXT I
190 GOSUB 1000: REM ESPERA
220 REM NORMALIZACION
230 FM = 0
240 FOR I=1 TO 10: IF F(I) > FM THEN FM=F(I)
250 NEXT I
260 FOR I=1 TO 10
270 FT(I) = F(I) * (20/FM)
280 NEXT I
290 REM GRAFICO
300 B=20
310 FOR I=1 TO 10
```

```
320 A = 21-FT(I): A = INT(A+0.5)
330 FOR Z=A TO B
340 LOCATE 4*I,Z
350 PRINT CHR$(143)
360 NEXT Z
370 NEXT I
380 PLOT 35,80: DRAW 630,80
390 PRINT: PRINT TAB(4) "EDADES EN CLASES DE 10 ANOS"
430 PRINT "FINAL": END
500 DATA 10.3,16.4,14.6,13.4,14.2,11.9,8.9,7.8,2.4,.2
1000 REM UP ESPERA
1010 LOCATE 5,23: PRINT "PULSAR UNA TECLA, POR FAVOR!"
1020 A$=INKEY$: IF A$="" THEN 1020
1030 CLS
1040 RETURN
```

5. PASO : Lista de variables

A = Límite inferior del grupo de edades
  después: dirección de inicio de la columna
B = Límite superior del grupo de edades
  después: dirección final de la columna
F = Frecuencia
FM = Frecuencia máxima
FT = Frecuencia transformada
I = Indice variable
Z = Indice variable (línea de pantalla)

6. PASO : Descripción del programa

Línea	10 - 90	Título
Línea	110	Espera
Línea	120 - 130	Dimensionado y lectura de datos iniciales
Línea	140 - 150	Visualización del encabezado de la tabla
Línea	160 - 180	Visualización de la tabla 165 : Determinación del punto inicial del grupo 170-176 : Visualización
Línea	190	Espera
Línea	220 - 250	Búsqueda de la frecuencia mayor y comparación de todas las frecuencias con cada frecuencia máxima alcanzada (comenzando por el valor 0)
Línea	260 - 280	Determinación de la frecuencia transformada mediante adaptación de las frecuencias a la frecuencia máxima
Línea	290 - 370	Trazado de 10 columnas 300 : Final de la columna 320 : Determinación del comienzo de la columna medido con respecto al reparto de líneas 330-360 : Trazado de la columna mediante el encadenamiento de pequeños cuadrados en la columna 3*I
Línea	380	Eje horizontal
Línea	390	Denominación del eje horizontal
Línea	430	Final del programa
Línea	500	Datos
Línea	1000 - 1040	Subrutina de espera

7. PASO : Resultados

No vamos a ofrecer los resultados de este programa. Nos parece mucho más útil que el lector introduzca el programa en su ordenador y que después intente explicar paso a paso lo que sucede, comparando las impresiones en pantalla con la descripción que hacemos.

No obstante, creemos conveniente hacer la siguiente puntualización:
El gráfico que traza este programa no utiliza las instrucciones BASIC que hay disponibles para los gráficos de alta resolución, sino que se genera mediante instrucciones PRINT que activan la función CHR$.

Esto tiene como consecuencia que el dibujo sea poco preciso. Las columnas de diagrama no representan exactamente las frecuencias, porque la unidad básica de trazado de las mismas es un cuadrado del tamaño del cursor, es decir con un lado de 7 mm. (en un monitor normal de ordenador).

Para obtener gráficos más exactos no queda más remedio que utilizar los gráficos de alta resolución, como haremos en el capítulo siguiente.

8.6 La economía de los paises europeos

1. PASO : Presentación del problema

El programa siguiente es muy distinto a los programas precedentes. Aquí se trata nuevamente de confeccionar un "programa informativo".

Vamos a desarrollar un programa que visualiza datos importantes sobre la economía nacional cuando el usuario lo requiere.

Estos datos se refieren a los siguientes campos:

1. Población (en millones)
2. Superficie (en miles de Km. cuadrados)
3. Producto interior bruto (en miles de millones de Ptas.)
4. Población activa (en millones)
5. Proporción de trabajadores extranjeros (en %)
6. Impuestos totales (en miles de millones de Ptas.)
7. Deudas (en miles de millones de Ptas.)
8. Porción de la producción industrial (en %)

El usuario puede alargar esta lista o utilizar otros conceptos en lugar de los países (p.e. Comunidades Autónomas).

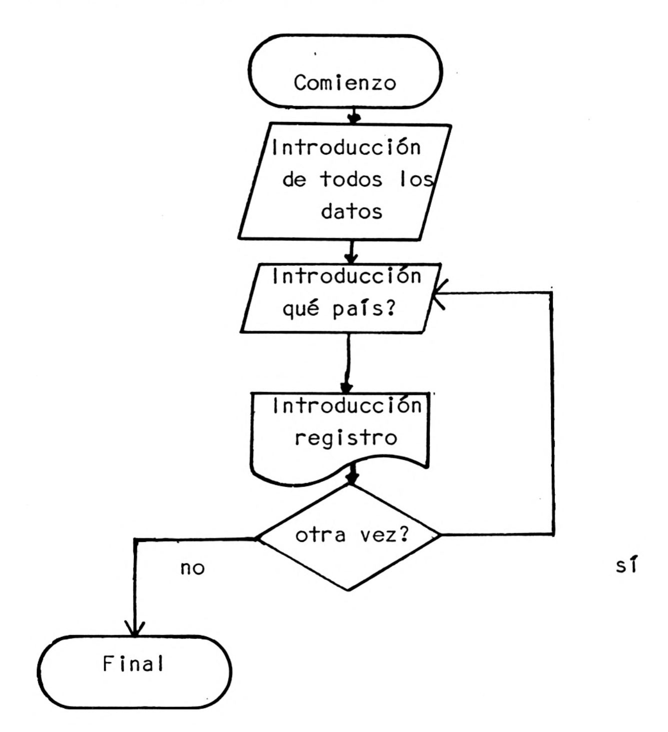
Lógicamente, habrá que modificar el fichero de datos para adaptar el programa.

2. PASO : Análisis del problema

El lector habrá podido comprobar que a lo largo de estos capítulos hay ciertos problemas que se repiten o reaparecen de forma similar a la originaria. Aquí nos encontramos de nuevo ante este problema: En principio no se presentan nuevos problemas y el análisis resulta ciertamente sencillo.

Primero hay que introducir todas las informaciones relevantes. A continuación se pregunta al usuario el nombre del país acerca del que desea obtener informaciones. Tras introducir este nombre tiene lugar la impresión del juego de datos correspondiente.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
10 REM E5 - PAISES
20 CLS
30 PRINT "PROGRAMA PARA IMPRIMIR INFORMACION SOBRE"
40 PRINT "EL PODER ECONOMICO DE DIVERSOS"
50 PRINT "PAISES EUROPEOS"
60 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS,1984"
70 PRINT: PRINT: PRINT: PRINT: PRINT
110 DIM L$(12), V(8,12), B$(8)
120 FOR I=1 TO 12: READ L$(I): NEXT I
130 FOR I=1 TO 8: READ B$(I): NEXT I
140 FOR J=1 TO 8: FOR I=1 TO 12: READ V(J,I): NEXT I: NEXT J
150 GOSUB 1000: REM ESPERA
200 PRINT "DE QUE PAIS DESEA "
210 PRINT "OBTENER INFORMACION? ": PRINT
220 FOR I=1 TO 12
230 PRINT TAB(5) "("; I; ")";: PRINT TAB(13) L$(I)
240 NEXT I
250 PRINT: PRINT: INPUT "INDIQUE NUMERO : "; Z
260 CLS
280 PRINT L$(Z): PRINT: PRINT
290 FOR J=1 TO 8
300 PRINT B$(J); TAB(29) ": ";: PRINT V(J,Z)
310 NEXT J
320 GOSUB 1000: REM ESPERA
330 PRINT "OTRO PAIS? (S/N) ";: INPUT A$
340 IF A$="S" THEN CLS: GOTO200
350 PRINT: PRINT "FINAL": END
500 DATA "ESPANA","FRANCIA","ITALIA"
502 DATA "REINO UNIDO","RFA","IRLANDA"
504 DATA "GRECIA","BELGICA","HOLANDA"
506 DATA "SUECIA","DINAMARCA","PORTUGAL"
508 DATA "NORUEGA","FINLANDIA","SUIZA"
509 DATA "LUXEMBURGO","AUSTRIA","MALTA"
```

```
510 DATA   POBLACION   (MILL.),SUPERFICIE   (1000   KM2),PROD.
INTERIOR(10E9)
512   DATA  POBLACION  ACTIVA  (MILL.),EXTRANJEROS   (%),IMPUESTOS
      TOTALES (10E9)
514 DATA DEUDAS (10E9),PROD. INDUSTRIAL (%)
600 DATA  0,0,0,0,0,0,0,0,0,0
602 DATA  0,0,0,0,0,0,0,0,0,0
604 DATA  0,0,0,0,0,0,0,0,0,0
606 DATA  0,0,0,0,0,0,0,0,0,0
608 DATA  0,0,0,0,0,0,0,0,0,0
610 DATA  0,0,0,0,0,0,0,0,0,0
612 DATA  0,0,0,0,0,0,0,0,0,0
614 DATA  0,0,0,0,0,0,0,0,0,0
616 DATA  0,0,0,0,0,0,0,0,0,0
618 DATA  0,0,0,0,0,0,0,0,0,0
620 DATA  0,0,0,0,0,0,0,0,0,0
622 DATA  0,0,0,0,0,0,0,0,0,0
624 DATA  0,0,0,0,0,0,0,0,0,0
626 DATA  0,0,0,0,0,0,0,0,0,0
628 DATA  0,0,0,0,0,0,0,0,0,0
1000 REM : ESPERA
1010 LOCATE 5,23: PRINT "PULSE UNA TECLA, POR FAVOR!"
1020 A$=INKEY$: IF A$="" THEN 1020
1030 CLS
1040 RETURN
```

5. PASO : Lista de variables

A$	Variable de cadena
B$	Denominación de variables
I	Indice variable
J	Indice variable (variables)
L$	Nombre de los países
V	Variables
Z	Número del país buscado

6. PASO : Descripción del programa

Línea 10 - 70	Título
Línea 110	Dimensionado
Línea 120 - 140	Lectura de los datos 120 : Nombre del país 130 : Denominación de las variables 140 : Valores de las variables
Línea 150	Espera
Línea 200 - 250	Demanda del número de país del cual se necesitan informaciones, después de ofrecer una lista de países en la pantalla
Línea 260 - 280	Borrado de la pantalla y visualización del nombre de país seleccionado
Línea 290 - 310	Visualización de las sentencias de datos del país seleccionado
Línea 320	Espera
Línea 330 - 350	Consulta si desea otra información (en caso afirmativo, borra la pantalla y bifurca a línea 200); en caso negativo, finaliza el programa.
Línea 500 - 508	Nombres de los países en las sentencias DATA
Línea 510 - 514	Sentencias DATA para denominar las variables
Línea 600 - 618	Valores de las variables. Introducir los valores de tal manera que para la primera variable se dan todos los valores del primer país, para la segunda todos los valores correspondientes al segundo, etc.

ATENCION!! EN EL PROGRAMA APARECEN UNICAMENTE CEROS (Ver siguiente paso)

Línea 1000 - 1040 Subrutina de pausa

7. PASO : Resultados

El fichero de datos de este programa contiene únicamente ceros, de manera que si hacemos una pasada de prueba, sólo aparecerán ceros en la pantalla.

Para efectuar una pasada " auténtica ", hay que introducir, primero, datos concretos, que el lector puede tomar de la tabla siguiente.

Datos económicos de diferentes países europeos :

País	Pobl. (mill.)	Superf. (1000 km2)	Produc. Interior (10E9)	Poblac. activa (mill.)
España				
Francia				
Italia				
Reino U.				
RFA				
Irlanda				
Grecia				
Bélgica				
Holanda				
Suecia				
Dinamarca				
Portugal				
Noruega				
Finlandia				
Suiza				
Luxemburgo				
Austria				
Malta				

País	Prop. extranjeros	Impues. totales (10E9)	Deudas. (10E9 Ptas.	Produc. indust. (%)
España				
Francia				
Italia				
Reino U.				
RFA				
Irlanda				
Grecia				
Bélgica				
Holanda				
Suecia				
Dinamarca				
Portugal				
Noruega				
Finlandia				
Suiza				
Luxemburgo				
Austria				
Malta				

Capítulo 9 : Matemáticas II



9.1 Consideración previa

Para finalizar incluimos un capítulo en el que se tratan nuevos problemas matemáticos, con un grado de dificultad algo mayor que los vistos hasta ahora. De esta forma, el usuario experimentado en la utilización del microordenador, comprobará que también se pueden abordar problemas de mayor complejidad con programas relativamente sencillos.
Llegados a este punto, podría parecer que el capítulo final se halla un poco por encima de las posibilidades del principiante (pero es precisamente esto lo que a veces le proporciona más interés).

Estos problemas de mayor complejidad, deben ser programados de tal forma que no requieran instrucciones BASIC adicionales. Esto significa, que los programas resultantes (al igual que todos los programas precedentes) no pueden satisfacer los niveles de elegancia de los programadores experimentados. Pero esto aquí no tiene importancia.

En contraposición a los anteriores capítulos, aquí seremos más breves, pues se van a tratar diecisiete nuevos problemas:

Después de una pequeña presentación del problema y de algunas indicaciones con referencia a su análisis, pasamos inmediatamente a presentar el programa y terminamos con una pequeña descripción del mismo. En adelante prescindiremos de los pasos "Diagrama de flujo" y "Resultados", no sólo por motivos de espacio, sino porque en cuanto al aspecto de la programación pura no introducen nuevos problemas de detalle.

9.2 Red

Presentación del problema

El programa siguiente sirve para trazar en la pantalla una red de líneas con separaciones entre líneas y columnas variable.
El esquema mental utilizado como base de estos programas, sirve de forma parecida por ejemplo para estructurar tablas extensas con líneas horizontales y verticales.

Análisis del problema

Podemos ahorrarnos cualquier explicación acerca del análisis del problema. El usuario únicamente debe introducir las distancias entre líneas y entre columnas. Estos datos sirven para posicionar el cursor gráfico por medio de instrucciones PLOT, y para luego trazar líneas rectas con DRAWTO.

Programa:

```
10 REM Z1-RED
20 CLS
30 PRINT "PROGRAMA PARA TRAZAR UNA RED"
40 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS,1984"
50 LOCATE 3,20: INPUT "DISTANCIA ENTRE COLUMNAS : "; S
60 LOCATE 3,22: INPUT "DÍSTANCIA ENTRE LINEAS   : "; Z
70 CLS
80 FOR I=1 TO 639 STEP S
90 PLOT I,399: DRAW I,1
100 NEXT I
110 FOR I=1 TO 399 STEP Z
120 PLOT 1,I: DRAW 639,I
130 NEXT I
140 LOCATE 18,24: PRINT "FIN": END
```

Lista de variables

I = Indice variable
S = Separación entre columnas
Z = Separación entre líneas

Descripción del programa

Línea 10 - 40	Título
Línea 50 - 60	Introducción de la distancia entre líneas y entre columnas.
Línea 70	Borrado de la pantalla
Línea 80 - 100	Trazado de las líneas verticales
Línea 110 - 130	Trazado de las líneas horizontales
Línea 140	Finalización del programa

9.3 La recta

Presentación del problema

El programa siguiente sirve para dibujar cualquier recta en un eje de coordenadas cuyo origen se sitúa en el centro de la pantalla (160,80).

Análisis del problema

La posición de una recta queda determinada por dos parámetros, la ordenada en el origen A y la tangente del ángulo que forma con el eje de ordenadas B, según la siguiente ecuación:

$$Y_i = A + B*X_i$$

Dado que el origen de coordenadas debe estar situado en (160,80), resulta que:

$$Y_i = 80 + A + B*(X_i - 160)$$

En todo caso, debemos tener en cuenta que en contraposición al eje de coordenadas normal, los valores de Y empiezan en la parte superior de la pantalla (la línea número 0 es la línea superior de la pantalla).

Por esta razón, debe modificarse la función de la forma siguiente:

$$Y_i = 160 - (80+A+B * (X_i -160))$$
$$= 80 - A-B * (X_i -160)$$

Programa:

```
10 REM Z2-RECTA
20 CLS
30 PRINT "PROGRAMA PARA TRAZAR UNA RECTA"
40 PRINT: PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS,1984"
50 LOCATE 3,20: INPUT "ABCISA EN EL ORIGEN A:  "; A
60 LOCATE 3,22: INPUT "PENDIENTE           B:  "; B
80 PLOT 0,200: DRAW 639,200
90 PLOT 320,399: DRAW 320,0
100 FOR X=0 TO 639
110 Y=200+A+B*(X-320)
120 IF Y <= 0 OR Y >= 399 THEN 140
130 PLOT X,Y
140 NEXT X
150 LOCATE 18,24: PRINT "FIN": END
```

Lista de variables

A = Abcisa en el origen
B = Tangente del ángulo de pendiente
X = Valores sobre el eje X
Y = Valores de la función

Descripción del programa

Línea	10 - 40	Título
Línea	50 - 60	Introducción de los parámetros de la recta
Línea	70	Borrado de la pantalla
Línea	80 - 90	Trazado de los ejes coordenados
Línea	100 - 140	Trazado de la recta sobre todo el ancho de la pantalla (si los valores de Y no son válidos se salta la instrucción 130 de dibujado)
Línea	150	Final del programa

9.4 El círculo

Presentación del problema

Siguiendo el mismo patrón del punto anterior, donde se desarrolló un programa para trazar una recta, de nuevo presentamos un programa para dibujar todo tipo de círculos. Aquí aparecen problemas ya tratados, de modo que seremos breves.

Análisis del problema

Nos enfrentamos aquí al mismo problema que en el punto anterior, con la diferencia de que aquí tomamos como base la ecuación del círculo:

Esta se expresa de la forma siguiente:

$$(x - a)^2 + (y - b)^2 = r^2$$

r es el radio del círculo, x e y son las coordenadas de la línea circular y a y b son las coordenadas del centro del círculo.

Si despejamos y (resolución de una ecuación cuadrática), obtenemos las ecuaciones para el semicírculo superior e inferior:

$$(y - b)^2 = r^2 - (x - a)^2 = D$$

$$y - b = \pm \text{ SQR(D)}$$

$$y_{1/2} = \pm b \quad \text{SQR(D)}$$

Con ello hemos aclarado los aspectos fundamentales de este programa:

Requiere como informaciones de entrada las coordenadas del centro del círculo a y b (que a partir de ahora llamaremos T y Z) y el radio r (en BASIC : R). A partir de aquí podrán determinarse los valores Y1 e Y2 correspondientes a diferentes valores de X y después trazar los puntos X,Y1 y X,Y2.

Debe prestarse atención a que X se mueva únicamente en el ámbito que va de T-R a T+R, porque fuera de esta zona no hay valores de Y definidos dentro del campo real (el valor auxiliar D sería negativo, y al tratar de extraer la raiz del mismo provocaría un mensaje de error y una interrupción del programa.

Programa

```
10 REM Z3 - CIRCULO
20 CLS
30 PRINT "PROGRAMA PARA TRAZAR UN CIRCULO"
40 PRINT: PRINT: PRINT: PRINT: PRINT
50 PRINT "ESTE PROGRAMA NECESITA "
60 PRINT "LAS INFORMACIONES INPUT "
60 PRINT " COLUMNA Y LINEA DEL "
70 PRINT "CENTRO DEL CIRCULO ASI "
80 PRINT "COMO SU RADIO"
90 PRINT: PRINT: PRINT: INPUT "COLUMNA DEL CENTRO : "; T
100 PRINT:              INPUT "LINEA DEL CENTRO   : "; Z
110 PRINT: PRINT:       INPUT "RADIO              : "; R
120 CLS
130 PLOT 0,0: DRAW 0,399
140 PLOT 0,0: DRAW 639,0
150 PLOT T,0: DRAW T,399
160 PLOT 0,Z: DRAW 639,Z
170 A=T-R: B=T+R
180 IF A < 0 THEN A=0
190 IF B > 639 THEN B=639
200 FOR X=A TO B
210 D=R*R-(X-T)^2
220 IF D <= 0 THEN Y=Z: GOTO 250
230 Y = Z+SQR(D)
240 IF Y < 0 OR Y > 399 THEN 260
250 PLOT X,Y
260 NEXT X
270 FOR X=B TO A STEP-1
280 D=R*R-(X-T)^2
290 IF D <= 0 THEN Y=Z: GOTO 320
300 Y=Z-SQR(D)
310 IF Y < 0 OR Y > 399 THEN 330
```

```
320 PLOT X,Y
330 NEXT X
340 LOCATE 18,24: PRINT "FIN": END
```

Lista de variables

A = Punto extremo izquierdo del círculo
B = Punto extremo derecho del círculo
D = D = $R^2 - (X\ T)^2$
I = Indice variable
R = Radio
T = Columna del centro del círculo
X = Valores de coordenada X
Y = Valores de coordenada Y
Z = Línea del centro del círculo

Descripción del programa

Línea 10 - 80	Título y aclaraciones
Línea 90 - 120	Informaciones input
Línea 130 - 140	Eje de coordenadas
Línea 150 - 160	Eje de coordenadas auxiliar
Línea 170 - 190	Fijación del ámbito de valores de X
Línea 200 - 260	Trazado de un semicírculo
Línea 270 - 330	Trazado del otro semicírculo
Línea 340	Final del programa

9.5 La sinusoide

Presentación del problema

En este último ejemplo de gráfico de alta resolución vamos a representar una oscilación sinusoidal, como la que ya nos encontramos en el capítulo ''Física''. El procedimiento a seguir es el mismo que en los dos puntos precedentes.

Análisis del problema

El análisis del problema es muy sencillo, y no se presentan cuestiones nuevas. Unicamente debe procurarse adaptar la periodicidad de la oscilación (frecuencia) y su amplitud (valor máximo) de forma que la representación gráfica en la pantalla sea ópticamente correcta, es decir, que el espacio disponible sea aprovechado de forma hábil.

Programa

```
10 REM Z4-SENO
20 CLS
30 PRINT "PROGRAMA PARA TRAZAR UNA SINUSOIDE"
40 PRINT: PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS,1984"
50 PRINT: PRINT: PRINT: PRINT
60 PRINT "EL PROGRAMA REQUIERE LAS SIGUIENTES "
70 PRINT "INFORMACIONES INPUT: ": PRINT: PRINT
80 PRINT: INPUT "FRECUENCIA (EN UNIDADES DE PI)  "; P
90 PRINT: INPUT "AMPLITUD (DE 0 A 199)           "; A
100 CLS
110 PLOT 0,200: DRAW 639,200
120 FOR X=0 TO 639
130 J=X/100: J=J*(1/P)
140 Y=SIN(J): Y=Y*A
150 Y=200+Y
160 PLOT X,Y
170 NEXT X
172 LOCATE 19,12: PRINT "PI"
174 LOCATE 38,12: PRINT "2PI"
176 LOCATE 3,21: PRINT "FRECUENCIA (EN PI) = "; P
178 LOCATE 3,22: PRINT "AMPLITUD           = "; A
180 LOCATE 18,24: PRINT "FIN": END
```

Lista de variables

A = Amplitud

J = Valor X transformado

P = Frecuencia

X = Valor de la abcisa X

Y = Valor de la ordenada Y

Descripción del programa

Línea 10 - 50	Título
Línea 60 - 90	Informaciones INPUT
Línea 100 - 110	Trazado de una línea central horizontal
Línea 120 - 170	Determinación de los valores de la función y trazado, después de transformar de acuerdo con P y A
Línea 172 - 180	Visualización del texto y final del programa

9.6 Tabla matemática 1

Presentación del problema

En el programa siguiente se imprimen los cuadrados y las raíces cuadradas de todos los números naturales del 1 al 100.
Este programa debe ser entendido casi como "programa de consulta".

Análisis del problema

Siendo tan sencillo el problema, pensamos que resulta superfluo cualquier análisis.

Programa

```
10 REM Z5 - TABLA
20 CLS
30 PRINT "PROGRAMA PARA GENERAR LOS CUADRADOS "
40 PRINT "Y LAS RAICES CUADRADAS DE TODOS LOS"
45 PRINT "NUMEROS NATURALES (1-100)."
50 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS, 1984"
60 GOSUB 2000: REM ESPERA
70 FOR I=1 TO 100
80 Q=I*I: W=SQR(I)
90 PRINT USING "###"; I;: PRINT TAB(7) USING "#####"; Q;:
   PRINT TAB(16) USING "###.###" ; W
100 IF I/15=INT(I/15) THEN GOSUB 2000: REM ESPERA
110 NEXT I
120 PRINT: PRINT "FINAL DE LA IMPRESION": END
1000 REM UP CABECERA
1010 PRINT "NUMERO    CUADRADO   RAIZ":PRINT
1020 FOR J=1 TO 23: PRINT "-";: NEXT J: PRINT: RETURN
2000 REM UP ESPERA
2010 LOCATE 5,23: PRINT  "PULSE UNA TECLA POR FAVOR!"
2020 A$=INKEY$: IF A$="" THEN 2020
2030 CLS
2040 GOSUB 1000: REM CABECERA
2050 RETURN
```

Lista de variables:

I = Indice variable (números naturales 1-100)
J = Indice variable
Q = Número cuadrado
W = Raiz cuadrada

Descripción del programa

Línea 10-50 : Título

Línea 60 : Salto a la subrutina 2000 para la espera

Línea 70-110 : Cálculo e impresión de los valores en cuestión

Línea 100: Después de cada 15 líneas salto a la subrutina 2000 para la espera

Línea 120 : Final del programa principal

Línea 1000-1020 : Subrutina para la impresión de la cabecera de la tabla

---

Línea 2000-2050 : Subrutina para la espera
Línea 2010 : Impresión de un mensaje
Línea 2020-2030 : Espera
Línea 2040-2050 : Borrado de la pantalla, impresión de una nueva cabecera mediante el salto al UP 1000 y después retorno al programa principal

9.7 Tabla matemática 2

Presentación del problema

Este programa sirve para ofrecer una tabla de valores de las funciones

- seno (SIN)
- coseno (COS)

Estas funciones angulares se representan en dependencia del parámetro (llamado pi en el programa). Por esto, en la tabla resultante aparecen también valores en grados angulares.

Análisis del problema

Tampoco aquí resulta necesario el análisis del problema, dado que no se presentan especiales dificultades.

Programa

```
10 REM Z6 - TABLA 2
20 CLS
30 PRINT TAB(11) "TABLA NUMERICA 2":PRINT PRINT
40 PRINT "PROGRAMA PARA DETERMINAR VALORES DE "
50 PRINT "FUNCIONES TRIGONOMETRICAS": PRINT
60 PRINT TAB(11) "SENO        (SIN)"
70 PRINT TAB(11) "COSENO      (COS)"
80 PRINT TAB(11) "TANGENTE    (TAN)"
85 PRINT: PRINT
90 GOSUB 2000 : REM ESPERA
100 FOR J=0 TO 40
102 I=J/10
105 A=(360*I)/(2*(22/7)): A=INT(A+0.5)
110 B=SIN(I)
120 C=COS(I)
130 D=TAN(I)
135 PRINT USING "≠≠.≠"; I;: PRINT TAB(6) USING "≠≠≠.≠≠≠"; A;
136 PRINT TAB(14) USING "≠≠≠.≠≠≠"; B;: PRINT TAB(21)
    USING "≠≠≠.≠≠≠"; C;: PRINT TAB(28) USING "≠≠≠.≠≠≠"; D
137 IF J=0 THEN 150
140 IF J/10 = INT(J/10) THEN GOSUB 2000
150 NEXT J
160 PRINT: PRINT "FINAL DE LA IMPRESION": END
1000 REM UP CABECERAS
1010 PRINT: PRINT "PI  GRADOS   SIN   COS   TAN": PRINT
1020 RETURN
2000 REM ESPERA
2010 LOCATE 5,23: PRINT "PULSE UNA TECLA, POR FAVOR!"
2020 A$=INKEY$: IF A$="" THEN 2020
2030 CLS
2040 GOSUB 1000
2050 RETURN
```

Lista de variables

A = Angulo (en grados)
A$ = Variable de cadena (respuesta)
B = Seno
C = Coseno
D = Tangente
I = J/10 (abcisa de la función angular)
J = Indice variable (aquí 0-40)

Descripción del programa

Línea 10 - 85 Título

Línea 95 Bifurcación a subrutina de espera (2000)

Línea 100 - 150 Cálculos e impresión de resultados.

Se determinan 41 valores (J=0 TO 40) siendo aplicadas las funciones angulares sobre I = J/10 (línea 102).

Línea 105-xxx : Transformación en grados

Línea 110-130 : Determinación del SIN, COS y TAN.

Línea 135-137 : Impresión

Línea 140 : Salto a la subrutina 2000 después de cada 10 vueltas (para evitar este salto cuando J=0, que aquí no tiene sentido, en la línea 137 para J=0 se salta a 150)

Línea 160 : Final del programa principal

Línea 1000-1020 : Subrutina para la impresión de la cabecera de la tabla

---

Línea 2000-2050 : Subrutina para la espera (descripción ver programa Z6-TABLA 1).

9.8 Tabla matemática 3

Presentación del problema

El programa siguiente sirve para imprimir los logaritmos neperianos ln (X) dentro del ámbito de valores de X que va de 1 a 100.

Análisis del problema

Debido a la sencillez del problema también prescindimos aquí de cualquier análisis.

Programa:

```
10 REM Z7 TABLA 3
20 CLS
30 PRINT "PROGRAMA PARA DETERMINAR "
40 PRINT "LOS VALORES DE LOS LOGARITMOS"
50 PRINT "NEPERIANOS"
60 PRINT: PRINT: PRINT TAB(9) "PROF.DR. W. VOSS,1984": PRINT
70 GOSUB 2000
80 FOR I=1 TO 100
85 L=LOG(I)
90 PRINT USING "####"; I;: PRINT TAB(8) USING "###.####"; L
100 IF I/15=INT(I/15) THEN GOSUB 2000
110 NEXT I
120 PRINT: PRINT: PRINT "FINAL DE LA IMPRESION": END
1000 REM UP CABECERAS
1010 PRINT: PRINT "NUMERO     LOGARITMO": PRINT
1020 RETURN
2000 REM UP ESPERA
2010 LOCATE 5,23: PRINT "PULSE UNA TECLA, POR FAVOR!"
2020 A$=INKEY$: IF A$="" THEN 2020
2030 CLS
2040 GOSUB 1000
2050 RETURN
```

Lista de variables :

I = Indice variable (aquí 1-100)

L = Logaritmo neperiano de I

Descripción del programa:

Línea 10-60 : Título

Línea 70 : Salto a la subrutina 2000 para la espera

Línea 80-110 : Cálculo de los logaritmos e impresión;

después de cada 15 líneas de la tabla esperar

Línea 120 : Final del programa principal

Línea 1000-1020 : Subrutina para la impresión de la cabecera de la tabla

---

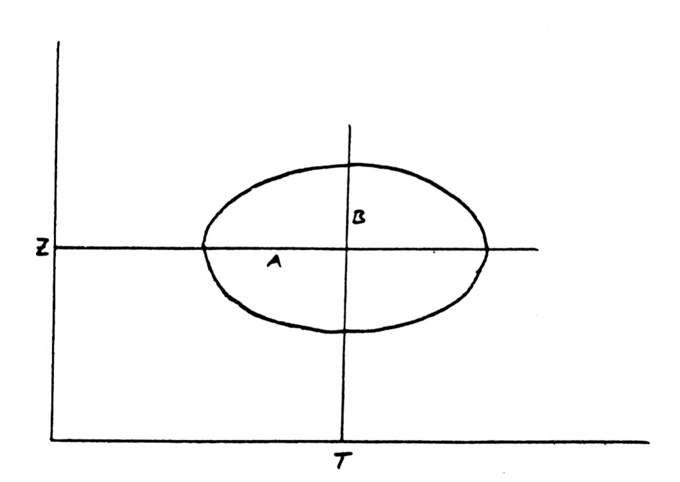
Línea 2000-2050 : Subrutina para esperar (descripción en el programa Z6-TABLA 1).

9.9 Elipse

Presentación del problema

El programa dibuja una elipse. Para ello deben introducirse los siguientes valores.

T: Columna del centro de la elipse (0-319)
Z: Línea del centro de la elipse (0-159)
A: Primer semieje de la elipse
B: Segundo semieje de la elipse



Análisis del problema

La ecuación de la elipse en la llamada forma central (centro de la elipse en el centro del eje de coordenadas):

$$\frac{x^2}{a} + \frac{y^2}{b} = 1$$

Para poder dibujar una función como esta mediante un programa BASIC, debemos despejar y, resultando:

$$y_{1/2} = \pm \frac{b}{a}\sqrt{a^2 - x^2}$$

Si tenemos en cuenta que una elipse no sólo puede tener su punto central en la coordenada (0,0), sino también en una cualquiera (T,Z), resultan necesarias correcciones mediante traslación del mismo (ver la siguiente descripción de programa).

Programa:

```
10 REM Z8-ELIPSE
20 CLS
30 PRINT "PROGRAMA PARA "
40 PRINT "TRAZAR UNA ELIPSE"
50 PRINT: PRINT: PRINT TAB(9) "PROF.DR. W. VOSS, 1984"
60 PRINT: PRINT: PRINT: PRINT
70 PRINT "ESTE PROGRAMA REQUIERE LAS SIGUIENTES"
80 PRINT "INFORMACIONES INPUT :": PRINT
90 PRINT "COORDENADAS DEL PUNTO CENTRAL  :"
100 PRINT TAB(5) ;: INPUT  "VALOR-X (COLUMNA)    : "; T
110 PRINT TAB(5) ;: INPUT  "VALOR Y (LINEA)      : "; Z
120 PRINT: INPUT "PRIMER SEMIEJE    A    : "; A
125 PRINT: INPUT "SEGUNDO SEMIEJE   B    : "; B
130 CLS
140 PLOT 0,0: DRAW 639,0
150 PLOT 0,0: DRAW 0,399
160 PLOT 0,Z: DRAW 639,Z
170 PLOT T,0: DRAW T,399
390 REM ELIPSE
400 C=T-A: D=T+A
410 IF C < 0 THEN C=0
420 IF D > 639 THEN D=639
430 FOR X=C+1 TO D-1
440 DS=(B/A)*SQR(A*A-(X-T)^2)
450 Y=Z+DS
460 IF Y < 0 OR Y > 399 THEN 480
470 PLOT X,Y
480 NEXT X
490 FOR X=D-1 TO C+1 STEP -1
500 DS=(B/A)*SQR(A*A-(X-T)^2)
```

```
510 Y=Z-DS
520 IF Y < 0 OR Y > 399 THEN 540
530 PLOT X,Y
540 NEXT X
550 LOCATE 5,23: PRINT "FINAL DE LA IMPRESION": END
```

Lista de variables

A = Primer semieje de la elipse
B = Segundo semieje de la elipse
C = Extremo izquierdo de la elipse
D = Extremo derecho de la elipse
DS = Valor auxiliar tomado de la ecuación de la elipse
T = Columna del punto central de la elipse
X = Abcisa
Y = Ordenada
Z = Línea del punto central de la elipse

Descripción del programa

Línea 10 - 60	Título
Línea 70 - 125	Fijación de las coordenadas del punto central y de los semiejes
Línea 130	Borrado de la pantalla
Línea 140 - 150	Trazado de los ejes coordenados
Línea 160 - 170	Trazado de los segundos ejes de coordenadas con centro en el punto central de la elipse.
Línea 390 - 480	Trazado de la mitad superior de la elipse
Línea 490 - 540	Trazado de la mitad inferior de la elipse
Línea 550	Final del programa

9.10 Distribución normal de Gauss

Presentación del problema

Este programa representa gráficamente la distribución de Gauss.
Tiene la función de densidad

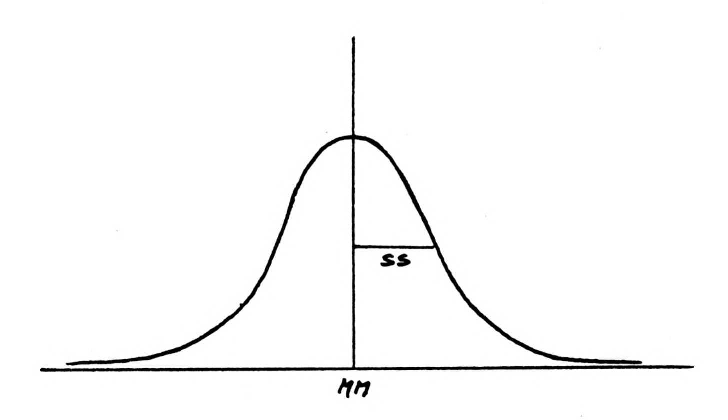
$$f(x) = \frac{1}{SS\sqrt{2\pi}} \cdot e^{\left[-\frac{1}{2}\left(\frac{X - MM}{SS}\right)^2\right]}$$

MM y SS son los parámetros de posición caracterizadores (ver el esquema siguiente):

MM designa el valor medio de la distribución
SS designa la dispersión de la distribución (desviación estandar = distancia entre la línea central y cualquiera de los dos puntos de inflexión de la distribución).

En este programa seleccionado los siguientes valores :

MM = 160
SS = 40



Análisis del problema

Nuevamente prescindible.

Programa:

```
10 REM Z9-GAUSS
20 CLS
30 PRINT "PROGRAMA PARA TRAZAR LA DISTRIBUCION"
40 PRINT "NORMAL SEGUN GAUSS"
50 PRINT: PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS,1984"
60 PRINT: PRINT: PRINT: PRINT
70 PRINT "ESTE PROGRAMA TRAZA LA DISTRIBUCION NORMAL"
80 PRINT "CON EL VALOR MEDIO 200 Y "
90 PRINT "LA DESVIACION ESTANDARD 60": PRINT
100 PRINT "ESTOS PARAMETROS PUEDEN MODIFICARSE"
110 PRINT "PARA CADA CASO (VER LINEA 120)"
115 GOSUB 1000: REM ESPERA
120 MM=200 : SS=60
130 CLS
140 PLOT 0,0: DRAW 400,0
150 PLOT 200,0: DRAW 200,300
390 REM DISTRIBUCION NORMAL
400 A=1/(SSP*SQR(2*(22/7)))
410 FOR X=0 TO 400
420 B=((X-MM)/SS)^2
430 B=0.5*B*(-1)
440 Y=A*EXP(B)
450 Y=(2000000/SS)*Y
460 IF Y<0 OR Y>399 THEN 480
470 PLOT X,Y
480 NEXT X
490 PRINT "FINAL DE LA IMPRESION": END
1000 REM UP ESPERA
1010 LOCATE 5,23: PRINT "PULSE UNA TECLA, POR FAVOR!"
1020 A$=INKEY$: IF A$="" THEN 1020
1030 CLS: RETURN
```

Lista de variables

A = Primer término de la ecuación funcional
A$ = Variable de cadena (respuesta)
B = Exponente de la ecuación funcional
MM = Valor medio de la distribución normal
SS = Desviación estandard de la distribución normal
X = Abcisa
Y = Ordenada

Descripción del programa

Línea 10 - 110	Título y aclaraciones
Línea 115	Espera
Línea 120 - 130	Definición de los parámetros de la función y borrado de la pantalla
Línea 140 - 150	Trazado de los ejes coordenados
Línea 390 - 480	Cálculo de los valores de ordenada de la distribución normal, normalización (línea 450) y trazado de la curva
Línea 490	Final del programa
Línea 1000-1030	Subrutina de espera

9.11 Regresión

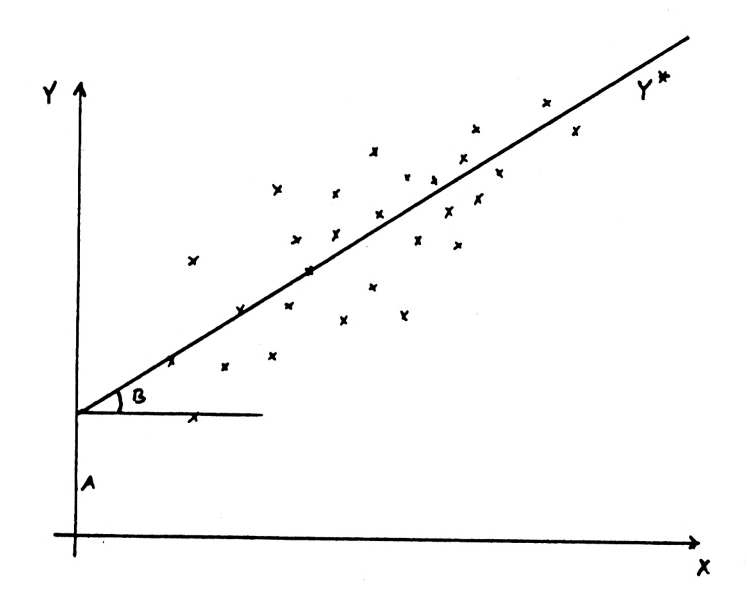
Presentación del problema

El cálculo de regresiones constituye una cuestión importante dentro del campo de la estadística matemática. Se trata de describir una distribución estadística bivariable mediante una función matemática.

Una distribución estadística bivariable se produce, por ejemplo, cuando estudiamos simultaneamente dos características de una población, es decir, si consultamos a una serie de personas acerca de su tamaño y de su peso al mismo tiempo.

El objetivo de una investigación como ésta, es descubrir si existe alguna relación entre ambas variables.

Una distribución bivariable puede representarse gráficamente en un sistema de coordenadas bidimensional por medio de una "nube de puntos", como puede apreciarse en la página siguiente.



Para describir el tipo de relación existente entre ambas variables, o para determinar la tendencia lineal en una serie de tiempos, que desde el punto de vista matemático es el mismo problema, se intenta encajar de la mejor forma posible una función matemática lineal entre los puntos.

Del mismo modo, existen directrices de cálculo no lineales, pero que aquí no serán tratadas.

Análisis del problema

La función lineal o no lineal que tratamos de dibujar se determina, por regla general, con el método de los mínimos cuadrados.

Este método nos da unas directrices de cálculo para determinar los parámetros de la función. En el caso de una recta, se trata de la ordenada en el origen A y de la pendiente B.

El método de los mínimos cuadrados nos exige determinar los parámetros de forma que la suma de los cuadrados de las desviaciones entre los valores de ordenada y (coordenadas verticales) observados y los valores de ordenada de la recta y deben ser minimizados, es decir:

$$\sum (y_i - \dot{y}_i)^2 = \min.$$

Los valores de $\dot{y}$ se calculan del modo siguiente con la ecuación de la recta:

$$y_i = a + b*x_i$$

Con la regla de minimización antes citada resulta:

$$\sum (y_i - a - b*x_i)^2 = min.$$

El mínimo de esta función se halla realizando una derivada parcial según a y b e igualando estas dos derivadas parciales a cero.

De esta forma se obtienen dos ecuaciones con dos incógnitas (a y b), en las que éstas se habrán de despejar. Se obtienen así las dos ecuaciones:

$$b = \frac{n*\sum x_i y_i - \sum x_i \sum y_i}{n*\sum x_i^2 - (\sum x_i)^2}$$

$$a = \frac{\sum y_i}{n} - b*\frac{\sum x_i}{n}$$

El programa BASIC a desarrollar sirve para calcular estos parámetros según el método de los mínimos cuadrados para cualquier par de valores x ,y que se introduzca y para a continuación imprimir los resultados.

Programa:

```
10 REM Z10 - REGRESION
20 CLS
30 PRINT "PROGRAMA PARA DETERMINAR LOS PARAMETROS "
40 PRINT "DE UNA FUNCION LINEAL DE REGRESION "
50 PRINT "O BIEN DE UNA RECTA DE TENDENCIA"
60 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS, 1984"
70 PRINT: PRINT: PRINT: PRINT: PRINT
80 PRINT "ESTE PROGRAMA REQUIERE PARES DE VALORES "
90 PRINT "COMO INFORMACIONES DE ENTRADA (POR EJEMPLO "
100 PRINT "PARES X E Y O DE TIEMPO Y CARACTERISTICA, ETC)"
110 GOSUB 1000: REM ESPERA
120 INPUT "CUANTOS PARES DE VALORES       : "; N
130 DIM X(N), Y(N)
140 PRINT: PRINT TAB(5) "INTRODUCCION    :   ": PRINT: PRINT
150 FOR I=1 TO N
160 PRINT I; ". PAR X,Y    :   ";: INPUT X(I),Y(I)
170 NEXT I
172 CLS: PRINT "CONTROL": PRINT: PRINT
174 PRINT "NRO.           VALOR X        VALOR Y": PRINT
176 FOR I=1 TO N: PRINT I,X(I),Y(I)
178 IF I/10=INT(I/10) THEN GOSUB 1000
179 NEXT I
180 REM REGRESION
185 GOSUB 1000: REM ESPERA
190 FOR I=1 TO N
200 S1=S1+X(I)*Y(I)
```

```
210 S2=S2+X(I): S3=S3+Y(I)
220 S4=S4+X(I)*X(I)
230 NEXT I
240 B=(N*S1-S2*S3)/(N*S4-S2*S2)
250 A=S3/N-B*S2/N
265 CLS
270 PRINT "RESULTADO    : ": PRINT: PRINT
280 PRINT          "ORDENADA EN EL ORIGEN A = "; A
290 PRINT: PRINT "PENDIENTE                B = "; B
300 PRINT: PRINT: PRINT: PRINT
310 PRINT "CON ESTOS PARAMETROS PUEDE LLAMARSE "
320 PRINT "AL PROGRAMA PARA EL TRAZADO DE UNA "
330 PRINT "RECTA SI DESEA OBTENERSE UNA "
340 PRINT "REPRESENTACION GRAFICA DEL RESULTADO"
350 PRINT: PRINT: PRINT "FINAL DE LA IMPRESION": END
1000 REM UP ESPERA
1010 LOCATE 5,23: PRINT "PULSE UNA TECLA, POR FAVOR!"
1020 A$=INKEY$: IF A$="" THEN 1020
1030 CLS
1040 RETURN
```

Lista de variables:

A = Ordenada en el origen de la recta

A$ = Variable de cadena (respuestas)

B = Pendiente de la recta

I = Indice variable

N = Cantidad de pares de valores

S1 =
S2 = Sumas auxiliares para calcular B y A
S3 =
S4 =

X = Abcisa

Y = Ordenada

Descripción del programa:

Línea 10-110	:	Título, aclaraciones y espera
Línea 120-130	:	Indicación del número de pares de valores de la característica y dimensionado
Línea 140-170	:	Introducción de los valores de la característica mediante INPUT
Línea 172-179	:	Impresión de control de los pares de valores y pausa después de cada diez pares
Línea 180-185	:	Espera
Línea 190-230	:	Formación de las cuatro sumas necesarias en las fórmulas para A y para B
Línea 240-250	:	Cálculo de los parámetros de la recta A y B
Línea 265-300	:	Impresión de los resultados
Línea 310-350	:	Indicación para bifurcar y final del programa

---

Línea 1000-1040 : Subrutina Espera

9.12 Permutaciones

Presentación del problema

El cálculo de permutaciones responde a la cuestión de cuántas formas distintas pueden ordenarse diferentes elementos.

Análisis del problema

Si por ejemplo tenemos tres elementos distintos a, b, y c, pueden distinguirse las siguientes posibilidades de ordenación:

abc, acb, bac, bca, cab, cba

Es decir, que con tres elementos se obtienen seis posibilidades distintas de ordenación.

Generalmente se cumple la siguiente regla:

Para el primero de los tres elementos existen tres posibilidades : puede estar en primer lugar, en segundo lugar o en tercer lugar;

entonces, para el segundo elemento existen solamente dos posibilidades cuando se ha decidido ya dónde debe colocarse el primer elemento.

Esto significa, no obstante, que los dos primeros juntos tienen 3*2=6 posibilidades.

Para el tercer elemento ya sólo queda una posibilidad y para los tres elementos 3*2*1 = 6 posibilidades distintas de ordenación.

De la misma forma, cuando se tienen n elementos distintos, resulta:

Número de posibilidades de ordenación =
n*(n-1)*(n-2)*(n-3)*...*3*2*1

Esta cadena de productos se resume mediante n! (n factorial), y se dice que n elementos pueden ordenarse de n factorial formas distintas, o :

que la permutación de n elementos distintos es n-factorial (n!).
Debe tenerse en cuenta que para valores de n elevados la expresión n! crece muy rapidamente. Es más, crece tan rápido, que el ordenador ya no puede representar el resultado de 34! a partir de n = 34.

Programa:

```
10 REM Z11 - PERMUTACIONES
20 CLS
30 PRINT "PROGRAMA PARA DETERMINAR EL NUMERO DE "
40 PRINT "PERMUTACIONES DE N ELEMENTOS "
50 PRINT "DIFERENTES (FACTORIAL N = N!)
60 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS, 1984"
70 PRINT: PRINT: PRINT: PRINT
80 INPUT "NUMERO DE ELEMENTOS N  :  "; N
90 P=N
100 IF N >= 34 THEN PRINT "N DEMASIADO GRANDE PARA ESTE CALCULO"
    : GOTO 150
110 FOR I=N-1 TO 1 STEP -1
120 P=P*I
130 NEXT I
140 PRINT: PRINT "FACTORIAL DE ";N; " ES "; P
150 PRINT: PRINT: PRINT "OTRO CALCULO (S/N) ";: INPUT A$
160 IF A$="S" THEN CLS: GOTO 80
170 PRINT: PRINT: PRINT "FINAL": END
```

Lista de variables:

A$ = Variable de cadena para respuesta
I = Indice (factor de los productos)
N = Número de elementos
P = Producto intermedio y final

Descripción del programa:

Línea 10-70 : Título y aclaraciones

Línea 80-90 : Demanda de la información INPUT (número de elementos a permutar. N) el campo del producto P toma el valor del campo, como primer factor de la cadena de productos a realizar

Línea 100 : Mensaje cuando N es igual o mayor a 34 porque la capacidad del ordenador no da para más, y salto a la línea 150

Línea 110-130 : Formación de la cadena de productos en el campo P

Línea 140 : Impresión del resultado

Línea 150-170 : Final del programa, a menos que se desee continuar calculando (entonces seguir en la línea 80 una vez borrada la pantalla)

9.13 Combinaciones

Presentación del problema

El problema es similar al del programa anterior, calculándose la cantidad de formas distintas en que pueden seleccionarse k elementos de entre n elementos distintos.

La fórmula de cálculo es la siguiente:

$$\text{Número} = \frac{n!}{k!(n-k)!}$$

A esta magnitud se le llama coeficiente binomial y se abrevia de la forma siguiente:

$$\binom{n}{k}$$

Por otra parte, éste equivale a los factores que resultan en el desarrollo de los llamados binomios:

$$(a+b)^2 = a^2 + 2ab + b^2$$

$$= 1a^2b^0 + 2a^1b^1 + 1a^0b^2$$

$$= \binom{2}{0}a^2b^0 + \binom{2}{1}a^1b^1 + \binom{2}{2}a^0b^2$$

O:

$$(a+b)^3 = a^3 + 3a^2b + 3ab^2 + b^3$$
$$= 1a^3b^0 + 3a^2b^1 + 3a^1b^2 + 1a^0b^3$$
$$= \binom{3}{0}a^3b^0 + \binom{3}{1}a^2b^1 + \binom{3}{2}a^1b^2 + \binom{3}{3}a^0b^3$$

Como puede observarse, todo binomio se descompone en sumandos. Siempre hay un sumando más que el indicado por el exponente del binomio (el exponente en el último ejemplo era 3; el número de sumandos 4).

Cada sumando consta de 3 factores :

- una constante (el coeficiente binomial que aquí nos ocupa)
- factor "a elevado..." con un exponente, que partiendo del exponente del binomio (aquí 3) baja hasta 0;
- factor b "b elevado a..." con un exponente, que partiendo de 0 sube hasta el exponente del binomio (aquí 3).

En el coeficiente binomial que nos ocupa, el mayor valor es el exponente del binomio (aquí 3); el valor menor es un número entero que va ascendiendo de derecha a izquierda desde 0 hasta el exponente del binomio.

El programa presentado determina cualquier coeficiente binomial, permitiéndonos calcular fácilmente también binomios más complejos (p.e. $(a+b)^{27}$).
Pero este programa puede utilizarse también para cálculos, como se demoostrará en los próximos programas.
Finalmente, un programa de este tipo, permite responder p.e. a la siguiente cuestión:

" 8 corredores están a punto de comenzar una carrera de 100 m lisos. Cuántas posibilidades diferentes existen de reparto de las tres primeras posiciones.?"

Respuesta:

$$\binom{8}{3} = \frac{8!}{3!(8-3)!} = \frac{8!}{3!5!} =$$

$$= \frac{8 \cdot 7 \cdot 6}{3 \cdot 2 \cdot 1} = \underline{\underline{56}}$$

Programa:

```
10 REM Z12-COMBINACIONES
20 CLS
30 PRINT "PROGRAMA PARA DETERMINAR EL NUMERO DE "
40 PRINT "COMBINACIONES DE N ELEMENTOS TOMADOS DE K EN K"
50 PRINT: PRINT: PRINT "SE REFIERE EN ESTE CASO AL "
55 PRINT "LLAMADO COEFICIENTE BINOMIAL"
60 PRINT: PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS,1984"
70 PRINT: PRINT: PRINT: PRINT
80 INPUT "NUMERO DE ELEMENTOS N     :  "; N
90 INPUT "NUMERO SELECCIONADO K     :  "; K
100 Y=N-K
110 IF K=0 OR K=N THEN BK=1: GOTO 210
120 IF K>N/2 THEN H=Y: Y=K: K=H
130 GOSUB 500
140 PX=A: A=N
150 IF PX=1 THEN PN=N: GOTO 200
160 FOR I=N-1 TO N-K+1 STEP -1
170 A=A*I
180 NEXT I
190 PN=A
200 BK=PN/PX
210 PRINT: PRINT: PRINT "EL NUMERO DE COMBINACIONES DE "; N
220 PRINT "ELEMENTOS TOMADOS DE  "; K; "ES DE   : "; BK
230 PRINT: PRINT: PRINT "FINAL DEL CALCULO": END
500 REM UP FAC
510 A=K
520 IF K=0 OR K=1 THEN A=1: GOTO 560
530 FOR I=K-1 TO 1 STEP -1
540 A=A*I
550 NEXT I
560 RETURN
```

Lista de variables:

A = Factorial (producto intermedio y final)
BK = Coeficiente binomial
H = Valor auxiliar en el cambio de K por Y
I = Indice variable (factor en el cálculo del factorial)
K = Número de elementos a elegir
N = Número de elementos
PN = Primer factorial en el coeficiente binomial
PX = Segundo factorial en el coeficiente binomial
Y = Y-K

Descripción del programa:

Línea 10-70 : Título y aclaraciones

Línea 80-90 : Introducciones (número de elementos N y número de elementos a elegir K)

Línea 100 : Definición del valor auxiliar Y

Línea 110 : Cálculo simplificado para K=0 ó K=N, porque por deficinición se cumple que

$$\binom{N}{0} = \binom{N}{N} = 1$$

Línea 120 : Intercambio de K y (N-K) (=Y) para K>N/2. Esto es posible porque se cumple que:

$$\binom{N}{K} = \frac{N!}{K!(N-K)!} = \binom{N}{N-K}$$

De este modo se simplifican los cálculos siguientes.

Línea 130 : Cálculo de K! en la subrutina 500

Línea 140-190 : Cálculo de otro factorial, que queda después de reducir (ver p.e. problema de los corredores de antes).

Línea 200-230 : Cálculo e impresión del coeficiente binomial y final del programa principal.

Línea 500-560 : Cálculo de K! en una subrutina

9.14 Lotería

Presentación del problema

Vamos a confeccionar un programa que calcula las posibilidades de pre io en la lotería (6 entre 49).

Análisis del problema

El programa calcula la probabilidad de conseguir 3, 4, 5 ó 6 aciertos en la lotería (6 de 49).

Las correspondientes probabilidades se calculan con la siguiente fórmula :

$$P(X) = \frac{\binom{6}{X} \cdot \binom{43}{6-X}}{\binom{49}{6}} = \frac{B1*B2}{B3}$$

X es el número de aciertos que debe indicar el usuario.

En suma, aquí tenemos que operar con tres coeficientes binomiales (B1,B2 y B3), que ya conocemos del capítulo anterior.
Por este motivo nos ahorramos ahora cualquier otra explicación.

Programa:

```
10 REM Z13-LOTERIA
20 CLS
30 PRINT "PROGRAMA PARA DETERMINAR "
40 PRINT "LA PROBABILIDAD DE ACIERTO DE"
50 PRINT " 3,4,5 O 6 NUMEROS EN LA LOTERIA (6 DE 49)"
60 PRINT: PRINT: PRINT: PRINT "PROF.DR.W. VOSS,1984"
70 PRINT: PRINT: PRINT: PRINT
80 PRINT "ESTE PROGRAMA NO CONSIDERA LA POSIBILIDAD "
90 PRINT "POSIBILIDAD DE ACERTAR 5 Y "
100 PRINT "EL NUMERO ADICIONAL!!"
110 GOSUB 1000: REM ESPERA
120 INPUT "CUANTOS ACIERTOS  "; X
125 IF X=6 THEN B1=1: B2=1: GOTO 300
130 REM B1
135 IF X=5 THEN B1=6: B2=43: GOTO 300
140 Y=6-X
150 P=X: FOR I=X-1 TO 1 STEP -1
160 P=P*I: NEXT I: P1=P
170 P=Y: FOR I=Y-1 TO 1 STEP -1
180 P=P*I: NEXT I
190 P2=P
200 B1=720/(P1*P2)
210 REM B2
220 X=Y
230 P=43
240 FOR I=P-1 TO 43-X+1 STEP -1
250 P=P*I: NEXT I
```

```
260 P1=P
265 P=X
270 FOR I=X-1 TO 1 STEP -1
280 P=P*I: NEXT I
290 P2=P: B2=P1/P2
300 REM B3
310 P1=49: P2=6
320 FOR I=1 TO 5
330 P1=P1*(49-I)
340 P2=P2*(6-I)
350 NEXT I
360 B3=P1/P2: W=(B1*B2)/B3
370 CLS
380 PRINT "     RESULTADO :": PRINT: PRINT
390 PRINT "LA PROBABILIDAD DE ACERTAR  ";X
400 PRINT "NUMEROS EN LA LOTERIA ES   : "
410 PRINT: PRINT: PRINT TAB(5) W
420 PRINT: PRINT: PRINT: PRINT "OTRO CALCULO (S/N) ": INPUT A$
430 IF A$="S" THEN CLS: GOTO 120
440 PRINT: PRINT: PRINT "FINAL DEL CALCULO": END
1000 REM UP ESPERA
1010 LOCATE 5,23: PRINT "PULSE UNA TECLA, POR FAVOR!"
1020 A$=INKEY$: IF A$="" THEN 1020
1030 CLS
1040 RETURN
```

Lista de variables:

A$ = Variable de cadena para introducir respuestas
B1 = Primer término en la fórmula de cálculo
B2 = Segundo término en la fórmula de cálculo
B3 = Tercer término en la fórmula de cálculo
P = Producto en el cálculo de factoriales
P1 = Primer factorial del coeficiente binomial
P2 = Segundo factorial del coeficiente binomial
W = Probalilidad
X = Número de aciertos
Y = 6-X

Descripción del programa:

Línea 10-100	: Título y aclaraciones
Línea 110	: Salto a la subrutina 1000 para la espera
Línea 120	: Introducción del número de aciertos
Línea 125	: Para X=6 se cumple que B1=B2=1 (ver programa Z10); seguir en la línea 310
Línea 130-200	: Determinación de B1 de acuerdo con las reflexiones hechas ya con referencia al programa Z10
Línea 210-290	: Del mismo modo, determinación del segundo coeficiente binomial B2
Línea 300-360	: Lo mismo con B3
Línea 360	: Cálculo de la probabilidad W
Línea 370-410	: Impresión del resultado
Línea 420-440	: Final del programa principal, a menos que se desee un nuevo cálculo
Línea 1000-1040	: Subrutina de espera

9.15 Distribución binomial

Presentación del problema

La distribución binomial es una distribución de probabilidades. Las distribuciones de probabilidades, a su vez, sirven para calcular -de la forma más sencilla posible- probabilidades, incluso en problemas complejos.

La distribución binomial es también muy importante porque sirve de "base" para toda una serie de procedimientos de cálculo de probabilidades.

La distribución binomial responde a la siguiente cuestión:

¿Cuál es la probabilidad de que en n intentos independientes entre sí se produzca x veces el suceso favorable, si en cada intento sólo son posibles dos resultados, y si la probabilidad de que se produzca este suceso favorable en un intento es del P %.

Sirva el siguiente ejemplo para ilustrar esta cuestión :

¿Cuál es la probabilidad de que echando 10 veces una moneda nos salga p.e. 5 veces cara?

Este es un caso de aplicación de la distribución binomial, pues en cada intento son posibles dos sucesos ("cara" y "cruz") y los intentos (las tiradas) son independientes entre sí.

Otro problema de este tipo sería :

¿Cuál es la probabilidad de que de cada 5 nacimientos 2 sean niña?

Análisis del problema

Para analizar el problema en cuestión nos ocuparemos en primer lugar del ejemplo citado en último lugar :

Si de cada cinco nacimientos dos deben ser niñas, la siguiente ''combinación de nacimientos'' podría responder al problema :

NA NA NO NO NO (NA = niña, NO = niño)

También sería válida la siguiente combinación :

NO NO NO NA NA

y también muchas otras.

El número total de combinaciones "favorables" puede hallarse con ayuda del cálculo de permutaciones, tal como ya comentamos.

Sin ser nuestra intención el demostrar este extremo, calculamos que el número de posibilidades de combinación de 2 niñas y 3 niños es:

$$\text{Número} = \frac{5!}{2! * 3!} = \frac{5*4*3*2*1}{2*1 * 3*2*1}$$

$$= \frac{5*4}{2*1} = 10$$

Generalmente, el número de disposiciones distintas de n elementos, de los que x son iguales (en el sentido de "favorables") y otros n-x son también iguales (en el sentido de "desfavorables"), es :

$$\text{Número} = \frac{n!}{x_i!*(n-x_i)!}$$

Para seguir con el análisis del problema nos basaremos en que la probabilidad de que nazca una niña es P = 0.5 - esta es la probabilidad de que se produzca un suceso favorable en un intento.

Contemplemos el resultado siguiente :

NA NA NO NO NO

¿Cuál es la probabilidad de que se produzca un resultado así en cinco intentos?

Contemplemos en primer lugar el nacimiento de una niña : La probabilidad de que se produzca es lógicamente 0.5.

La probabilidad de que los dos primeros nacimientos juntos sean niña es entonces 0.5 * 0.5 (en general P * P).

La probabilidad de que nazca un niño es 1 - 0.5 (en general 1 - P). De esta manera, siguiendo el mismo procedimiento de arriba, la probabilidad de que el tercero, el cuarto y el quinto nacimiento sean niño es :

$$(1 - 0.5)*(1 - 0.5)*(1 - 0.5)$$

o, en general :

$$(1 - P)*(1 - P)*(1 - P)$$

Para la combinación NA NA NO NO NO obtenemos en total la probabilidad siguiente (escrita de forma general) :

$$P * P * P\ (1 - P)*(1 - P)*(1 - P)$$
$$= P^{2} * (1 - P)^{3}$$
$$= P^{x_i} * (1 - P)^{n-x_i}$$

En nuestro ejemplo obtenemos :

Probabilidad para NA NA NO NO NO =

$$0.5^{2} * (1 - 0.5)^{5-2} = 0.5^{5} = 1/32$$

Pero, en el ejemplo que nos ocupa hay 10 posibles combinaciones distintas NA,NO .

Así a partir de 10 posibles combinaciones, con una probabilidad de 1/32 cada una, se obtiene :

Probabilidad de que de 5 nacimientos 2 sean niña = 10 * 1/32 = 10/32 = 31.25 %

Este ejemplo nos permite inducir la siguiente fórmula general :

La probabilidad de que en n intentos se produzca x veces el suceso favorable, cuando la probabilidad individual de producirse un suceso es P, se calcula del modo siguiente :

$$W = \frac{n!}{x_i! * (n-x_i)!} * P^{x_i} * (1-P)^{n-x_i}$$

Esta fórmula parece complicada, pero su programación resulta relativamente sencilla.

Programa:

```
10 REM Z14 - BINOMIAL
20 CLS
30 PRINT "PROGRAMA  PARA  CALCULAR  "
40 PRINT "LAS  PROBABILIDADES  CON  LA "
50 PRINT "DISTRIBUCION BINOMIAL"
60 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS, 1984"
70 PRINT: PRINT: PRINT: PRINT
80 PRINT  "CON  ESTE  PROGRAMA, ES DECIR CON "
90 PRINT  "LA  DISTRIBUCION BINOMIAL PUEDE "
100 PRINT "RESPONDERSE A LA SIGUIENTE "
110 PRINT "CUESTION:  CUAL ES LA "
120 PRINT "PROBABILIDAD  DE PRODUCIRSE "
130 PRINT "X  VECES EL SUCESO FAVORABLE "
140 PRINT "EN  N INTENTOS, SI LA PROBABILIDAD "
150 PRINT "EN UN INTENTO ES P ? "
160 GOSUB 1000: REM ESPERA
170 PRINT "ESTE PROGRAMA NECESITA LAS SIGUIENTES "
180 PRINT "INFORMACIONES  DE ENTRADA : ": PRINT
190 PRINT: INPUT "NUMERO DE INTENTOS              : "; N
200 PRINT: INPUT "NUMERO DE SUCESOS FAVORABLES    : "; X
220 PRINT: PRINT  "PROBABILIDAD  EN  UN  INTENTO  "
230 INPUT "(COMO  NUMERO DECIMAL)        : "; P
240 REM COEFICIENTE BINOMIAL
245 Y=N-X: Q=1-P
```

```
250 IF X=0 OR X=N THEN BK=1: GOTO 360
260 IF X > N/2 THEN H=Y: Y=X: X=H: G=Q: Q=P: P=G
270 F=X: GOSUB 500: REM FAC
290 PX=A: A=N
295 IF PX=1 THEN PN=N: GOTO 350
300 FOR I=N-1 TO N-X+1 STEP -1
310 A=A*I
320 NEXT I
330 PN=A
350 BK=PN/PX
360 W=BK*P^X*Q^Y
365 CLS
370 PRINT "     RESULTADO  : ": PRINT: PRINT
380 PRINT "LA PROBABILIDAD DE PRODUCIRSE EN "; N
390 PRINT " INTENTOS "; X
400 PRINT " VECES EL SUCESO FAVORABLE ES   : "
410 PRINT: PRINT: PRINT TAB(5) W
420 PRINT: PRINT: PRINT "OTRO CALCULO (S/N) ?";: INPUT A$
430 IF A$ = "S" THEN CLS: GOTO 190
440 PRINT: PRINT: PRINT "FIN DEL CALCULO": END
500 REM UP FAC
510 A=X
520 IF X=0 OR X=1 THEN A=1: GOTO 560
530 FOR I=X-1 TO 1 STEP -1
540 A=A*I
550 NEXT I
560 RETURN
1000 REM UP ESPERA
1010 LOCATE 5,23: PRINT "PULSE UNA TECLA, POR FAVOR!"
1020 A$=INKEY$: IF A$="" THEN 1020
1030 CLS
1040 RETURN
```

Lista de variables:

A = Producto en el cálculo de factoriales
A$ = Variable de cadena (respuestas)
BK = Coeficiente binomial
F = Valor auxiliar (F=X)
G = Valor auxiliar en el cambio de P y Q
H = Valor auxiliar en el cambio de X e Y
I = Indice variable
N = Número de intentos
P = Probabilidad de producirse el suceso favorable en cada intento
PN = Primer término del coeficiente binomial
PX = Segundo término del coeficiente binomial
Q = 1-P
W = Probabilidad
X = Número de sucesos favorables
Y = N-X

Descripción del programa

Línea 10-150 : Título y aclaraciones

Línea 160-230 : Espera e introducción de las informaciones INPUT (número de intentos, probabilidad).

Línea 240-350 : Determinación del coeficiente binomial :

245 : Campos auxiliares;
250 : Si X=0 ó X=1 entonces el coeficiente se define como 1; seguir en 360;
260 : Si X>N/2 entonces, para aprovechar las posibilidades de reducción, cambiar X por N-X y P por 1-P
270 : Definición del denominador (=PX) del coeficiente y redefinición de A,en caso de utilizar nuevamente var. A (300);
295 : Si el denominador del coeficiente es = 1, entonces el numerador queda definido como N; seguir en línea 350.
300-320 : Determinación del numerador del coeficiente BK;
350 : Cálculo de BK.

Línea 360 : Cálculo de la probabilidad buscada.

Línea 365-410 : Impresión del resultado del cálculo.

Línea 420-499 : Final del programa, a menos que no se quiera un nuevo cálculo (entonces volver a la línea 190 tras borrar la pantalla).

Línea 500-560 : Subrutina para calcular un factorial

Línea 1000-1040 : Subrutina para la espera.

9.16 Teorema central de los límites (TCL)

Presentación del problema

Supongamos una muestra, que se toma al azar, de amplitud 50 a partir de una totalidad (por ejemplo se les consulta a 50 alumnos elegidos en una escuela, acerca de si están contentos con sus profesores).

En una muestra como ésta pueden calcularse varias magnitudes estadísticas, como por ejemplo la media aritmética (edad promedio de los alumnos o similares).

Si ahora tomásemos al azar todas las muestras distintas de amplitud n=50 posibles (nadie hará esto en la práctica), obtendremos muchos valores medios. Si dibujamos una distribución de frecuencias de estos valores medios, obtendremos una distribución que se ajusta de forma muy precisa al modelo de la distribución normal, independientemente de si la variable que nos ocupa (p.e. la edad de los alumnos) está repartida en la totalidad inicial de la que hemos tomado la muestra.

De esta forma puede pronosticarse la probabilidad con que aparece un valor medio a partir de una sóla muestra tomada al azar (y esto es lo que ocurre en la práctica).

Podemos calcular entonces dentro de qué límites debe situarse la media desconocida de la totalidad, indicando una probabilidad concreta (que en determinados casos puede ser bastante alta; p.e. el 90, 95 ó 99 %), una vez se ha determinado un valor medio en una muestra.

Estos cálculos de probabilidades son posibles únicamente porque los podemos basar en la validez de la distribución normal, como se demuestra en el Teorema Central de los Límites de LaPLace y Liapunoff :

Los parámetros de las muestras tomadas al azar se ajustan aproximadamente a la distribución normal de Gauss, independientemente del como es cada distribución en la totalidad base.

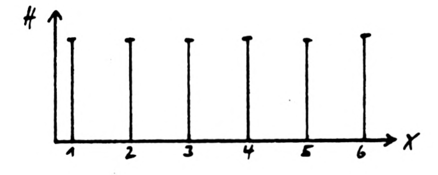
El siguiente programa demuestra la validez del teorema mediante un sencillo ejemplo. Nos enseña que los valores medios de las muestras tomadas al azar están distribuidos de una forma aproximadamente normal, a pesar de que la totalidad base de la que se extraen estas muestras, no es en absoluto "normal" (en el sentido de Gauss).

Para representar el teorema de los límites, hacemos que el ordenador extraiga 80 muestras al azar, cada una de ellas con una amplitud de n = 50. La variable objeto de este estudio es la tirada de un dado.

Esto significa que dejamos que el ordenador eche 50 veces un dado, calculamos el promedio de puntos de esta muestra; dejamos al ordenador echar otras 50 veces el dado y calculamos el promedio de puntos y, así, de 50 en 50 veces hasta alcanzar un total de 80 muestras.

Después de esta operación disponemos de 80 promedios que a su vez, pueden ser representados en forma de distribución de frecuencias por parte del ordenador.

La impresión de esta distribución de frecuencias demuestra que la misma ya no se ajusta, ni mucho menos, al modelo de una distribución de puntos, que debería presentar el siguiente aspecto :



Todo lo contrario : Obtenemos una distribución normal (además, esta similitud con el modelo de la distribución normal de Gauss es tanto mayor, cuanto mayor es el número de muestras tomadas al azar).

Además de la representación gráfica de la distribución de frecuencias resultante, calculamos también los parámetros caracterizantes, es decir, la media aritmética y la desviación estándar. Se cumple que :

$$\text{Media aritmética} = \frac{\sum x_i}{n}$$

(del mismo modo puede definirse también un valor medio de todas las medias de las muestras);

$$\text{Desviación estandar} = \sqrt{\frac{1}{n}\sum (x_i - x)^2}$$

(x es cada una de las medias aritméticas; del mismo modo puede definirse también una desviación estándar de todas las medias de las muestras).

En suma, los pasos a seguir son los siguientes :

1. En el programa a confeccionar deben simularse 80 veces 50 tiradas de dado, y de cada serie de 50 debe calcularse la media aritmética.

2. Además se debe calcular la media total y como magnitud de dispersión de las 4000 tiradas la desviación estándar.

3. En un tercer paso se confeccionará la distribución de frecuencias de las 80 medias obtenidas, sorteando estos valores mediante una pauta de clasificación. Cada una de las clases tendrá un ancho de 0.1 unidades (este es el ancho por clase más adecuado).

4. A continuación, se determinarán la media y la desviación estándar de esta distribución de frecuencias, y ambos valores se compararán con los correspondientes valores obtenidos en el punto 2.

5. Finalmente, se representará gráficamente la distribución de frecuencias de las medias de las muestras a efectos de una mayor claridad.

Programa:

```
10 REM Z15-TCL
20 CLS
30 PRINT "PROGRAMA  PARA  ILUSTRAR EL "
40 PRINT " TEOREMA  CENTRAL  DE  LOS "
50 PRINT "LIMITES (DISTRIBUCION NORMAL DE GAUSS)"
60 PRINT: PRINT: PRINT TAB(9) "PROF.DR.W. VOSS, 1984"
70 PRINT: PRINT: PRINT: PRINT: PRINT
80 PRINT  "ESTE PROGRAMA REALIZA 100 SERIES "
90 PRINT  "DE 50 TIRADAS CADA UNA;  CALCULA "
100 PRINT "LA MEDIA DE CADA SERIE,  A PARTIR  "
110 PRINT "DE  LA  CUAL REALIZA UNA DISTRIBUCION"
120 PRINT " DE FRECUENCIAS"
122 GOSUB 1100: REM ESPERA
125 N=100
130 DIM AM(N), KA(11), KE(11), H(11), W(N,50)
140 FOR I=' TO N
150 S=0
160 FOR J=1 T[illegible]
170 W(I,J)=INT[illegible](1)*6+1)
180 S=S+W(I,J): SS=SS+W(I,J)
190 NEXT J
200 AM(I)=S/50
210 NEXT I
215 MW=SS/(N*50): PRINT:
    PRINT "MEDIA GENERAL = ";MW: PRINT: PRINT
```

```
216 GOSUB 1100:REM ESPERAR
217 FOR I=1 TO N:FOR J= 1 TO 50
218 S1=S1+(W(I,J)-MW)^2
219 NEXT J: NEXT I
223 S9=SQR(S1/(N*50))
225 PRINT: PRINT "DESVIACION ESTANDARD    =  ";S9
226 PRINT: PRINT: PRINT: PRINT: PRINT: PRINT
227 REM DISTRIBUCION DE FRECUENCIAS
230 GOSUB 1000: REM STOP
240 PRINT "FRECUENCIA MEDIA EN SERIES DE 50     "
250 PRINT
260 FOR I=1 TO 30: PRINT "-";: NEXT I: PRINT
270 FOR K=1 TO 10
280 KA(K)=3+(K-1)*0.1
290 KE(K)=KA(K)+0.1
300 NEXT K
310 FOR I=1 TO N
320 FOR K=1 TO 10
330 IF AM(I) < KE(K) THEN H(K)=H(K)+1: GOTO 350
340 NEXT K
345 H(11)=H(11)+1
350 NEXT I
360 FOR K=1 TO 10
370 PRINT KA(K); TAB(6) "HASTA MENOS DE "; KE(K); TAB(25) H(K)
380 NEXT K
385 PRINT TAB(11) "MAS DE "; KE(10); TAB(25) H(11)
390 REM MEDIA DE LA DISTRIBUCION
```

```
400 S=0
410 FOR J=1 TO N
420 S=S+AM(J)
430 NEXT J
440 A=S/N
450 PRINT: PRINT: PRINT "MEDIA DE LA DISTRIBUCION  : "; A
455 REM DISPERSION DE LA DISTRIBUCION
460 S=0
470 FOR J=1 TO N
480 S=S+(AM(J)-A)^2
490 NEXT J
500 SS=SQR(S/N)
510 PRINT: PRINT: PRINT "DISPERSION DE LA DISTRIBUCION   : ";:
    PRINT USING "≠≠≠.≠≠"; SS
515 GOSUB 1000: REM ESPERA
520 PRINT: PRINT: PRINT "   COMPARACION    :   ": PRINT: PRINT
530 PRINT TAB(20) "DADO    GAUSS": PRINT
560 PRINT "MEDIA";: PRINT TAB(20) USING "≠≠.≠≠"; Mw;:
    PRINT TAB(30) USING "≠≠.≠#"; A
570 PRINT: PRINT "DISPERSION";: PRINT TAB(20) USING "≠≠.≠≠"; S9;:
    PRINT TAB(30) USING "≠≠.≠≠"; SS
580 GOSUB 1000: REM STOP
```

```
600 REM GRAFICO
610 PLOT 0,80
620 DRAW 600,80
630 FOR K=1 TO 11
640 E=20: A=E-H(K): IF A<3 THEN A=3
645 LOCATE 3*K-1, A-2: PRINT H(K)
650 FOR J=A TO E
660 LOCATE 3*K,J: PRINT CHR$(143)
670 NEXT J
680 NEXT K
730 LOCATE 5,23
740 PRINT "FINAL DE LA IMPRESION": END
1000 REM UP STOP
1010 LOCATE 5,23: PRINT "PULSE UNA TECLA, POR FAVOR!"
1020 A$=INKEY$: IF A$="" THEN 1020
1030 CLS
1040 RETURN
1100 REM UP ESPERA
1110 PRINT:PRINT:PRINT TAB(7) "POR FAVOR ESPERE, ESTOY CALCULANDO"
1120 RETURN
```

Lista de variables:

A = Media de todas las medias, después dirección inicial para dibujar
AM = Media de las muestras
A$ = Variable de cadena (respuestas)
E = Dirección final para dibujar
H = Frecuencia
I = Indice variable (serie número)
J = Indice variable (tirada número)
K = Indice variable (clase)
KA = Punto inicial de una clase
KE = Punto final de una clase
MW = Media total
N = Número de tiradas
S = Suma de cada serie
SS = Suma total
S1 = Suma de cuadrados
S9 = Desviación estandard
W = Tirada

Descripción del programa:

Línea 10-120 : Título y aclaraciones.

Línea 122 : Espera (mediante la subrutina 1100).

Línea 125-130 : Indicación del número de muestras a extraer y dimensionado de los campos para variables requeridos (se parte de una amplitud de las muestras de 50).

Línea 140-210 : Generación de N muestras de 50 tiradas cada una :

150 : Puesta a cero del campo para sumas S;
160-190 : Se echan 50 veces los dados.
170 : Tirada de un dado;
180 : Cálculo de la suma de puntos de cada muestra S y de la suma total SS;
200 : Determinación de la media de la muestra AM.

Línea 215-216 : Cálculo de la media total MW e impresión de la misma.

Línea 217 : Espera.

Línea 218-225 : Cálculo de la desviación estándar total como magnitud de dispersión de los puntos e impresión de este valor.

219 : Suma de los cuadrados de todas

las desviaciones de los puntos con respecto a la media total que ocupa el campo S1;

223 : Cálculo de la desviación estándar;

225 : Impresión del resultado.

Línea 227-230 : Espera.

Línea 240-260 : Impresión de una cabecera de la tabla para la distribución de frecuencias de las medias.

Línea 270-300 : Determinación de los límites inferiores (KA) y de los límites superiores (KE) de las clases para la distribución de frecuencias a confeccionar (ancho de las clases=0.1).

Línea 305 : Puesta a cero de los campos para las frecuencias.

Línea 310-350 : Ordenación de las medias por clases de frecuencias;

310 : Fijación de uno de los valores medios (del total de N valores medios);
320 : Fijación de una clase de frecuencia (del total de 10 clases);
330 : Si la media es menor que el límite superior KE de la clase incluirla en esta clase (H=H+1); seguir en 350 (siguiente media); en caso contrario seguir en 340 (siguiente clase);
340 : siguiente clase;
345 : ocupación de una clase residual;
350 : siguiente media

Línea 360-385 : Impresión de la distribución de frecuencias

Línea 390-450 : Cálculo de la media de todas las medias (= media de la llamada distribución de muestras) :

400 : Campo para la suma S = 0;
410-430 : Suma acumulativa de todas las medias AM;
440 : Cálculo de la media total A;
450 : Impresión de A.

Línea 455-510 : Cálculo de la dispersión de todas las muestras (= el llamado error de las muestras) :

460 : Campo para la suma S = 0;
470-490 : Cálculo de la suma de los cuadrados de todas las desviaciones entre las medias de las muestras y la media total;
500 : Cálculo de la desviación estándar (= dispersión de la distribución de muestras);
510 : Impresión.

Línea 515 : Espera

Línea 520-570 : Comparación de las magnitudes entre sí

Línea 580 : Espera

Línea 600-700 : Generación de un gráfico de la distribución de frecuencias:

630 : Fijación de una clase de frecuencias;

640 : Determinación del punto final y del punto inicial de la barra que representa una de las frecuencias;
645 : Represión si H=0
650-670 : Trazado de la barra
680 : Siguiente clase

Línea 740 : Final del programa

Línea 1000-1040 : Subrutina de espera

Línea 1100-1120 : Subrutina mensaje de espera

Una vez iniciada la ejecución de este programa, se requieren aprox. 3 minutos para producir el primer resultado aleatorio :

MEDIA TOTAL = 3.455 (depende del azar)

Después de otros 15 minutos, el ordenador informa :

DESVIACION ESTANDAR = 1.676 (depende del azar)

Si el dado es completamente correcto, la media esperada debe situarse en 3.5 y la desviación estándar esperada en 1.708.

A partir de ello se obtiene la siguiente distribución de frecuencias, que representa un suceso al azar :

MEDIA DE LAS SERIES DE 50	NUMERO
3 HASTA MENOS DE 3.1	5
3.1 HASTA MENOS DE 3.2	8
3.2 HASTA MENOS DE 3.3	7
3.3 HASTA MENOS DE 3.4	11
3.4 HASTA MENOS DE 3.5	12
3.5 HASTA MENOS DE 3.6	10
3.6 HASTA MENOS DE 3.7	13
3.7 HASTA MENOS DE 3.8	9
3.8 HASTA MENOS DE 3.9	2
3.9 HASTA MENOS DE 4	2
MAS DE 4	1

MEDIA DE LAS DISTRIBUCION : 3.45
DISPERSION DE LA DISTRIBUCION : .25

A continuación, el ordenador imprime :

COMPARACION

	DADO	GAUSS
MEDIA	3.46	3.45
DISPERSION	1.68	0.25

Observamos que la dispersión de la distribución de las muestras, que - tal como ya comentamos - se asemeja a la distribución normal de Gauss, es mucho menor que la de la distribución inicial. En este extremo se basa a fin de cuentas la precisión que pueden alcanzar los cálculos por encima realizados partiendo de muestras al azar.

El teorema central de los límites demuestra, entre otras, las siguientes relaciones :

Media de la distribución de las muestras	$\sim$	Media de la totalidad base
3.45	$\sim$	3.46
Dispersión de la distribución de las muestras	$\sim$	Dispersión de la totalidad base dividida entre la raíz de la población de las muestras
0.25	$\sim$	$1.68/\sqrt{50}$
	=	1.68/7.07...
	=	0.248

Finalmente, el programa produce una representación gráfica de la distribución de frecuencias de las medias antes ofrecidas, donde las diversas frecuencias aparecen en forma de barras de diferentes longitudes.

9.17 Elecciones

Presentación del problema

El programa representa gráficamente, en forma de un diagrama de barras, los resultados de las elecciones.

Para ello, el usuario debe introducir el porcentaje de votos obtenido por los partidos señalados en el programa : PSOE, AP, CIU, PCE y otros.

Análisis del problema

Dada la sencillez del problema podemos prescindir del análisis del mismo.

Programa:

```
10 REM Z16-ELECCIONES
20 CLS
30 PRINT "PROGRAMA PARA LA REPRESENTACION GRAFICA DE "
40 PRINT "RESULTADOS ELECTORALES"
50 PRINT: PRINT: PRINT: PRINT
60 PRINT TAB(9) "PROF.DR. W. VOSS, 1984"
70 PRINT: PRINT: PRINT
80 PRINT "INTRODUCIR LOS RESULTADOS ELECTORALES EN %"
90 PRINT: PRINT: PRINT
95 DIM F(5), G(5)
100 INPUT "PSOE       : "; F(1)
110 PRINT "AP         : "; F(2)
120 PRINT "CDC        : "; F(3)
130 PRINT "PCE        : "; F(4)
140 PRINT "OTROS      : "; F(5)
150 CLS
160 PLOT 0,110: DRAW 600,110
180 FM=0: FOR I=1 TO 5: IF F(I) > FM THEN FM = F(I)
190 NEXT I
200 FOR I=1 TO 5: G(I) = (17/FM)*F(I)
210 G(I)=INT(G(I)+0.5): NEXT I
```

```
230 GOSUB 2000: REM DIBUJAR
250 LOCATE 1,22
260 PRINT "PSOE     AP      CDC      PCE      OTROS"
270 FOR J=1 TO 5: PRINT TAB(6*(J-1)) F(J);: NEXT J
280 PRINT: PRINT "FINAL": END
2000 REM UP DIBUJAR
2010 FOR J=1 TO 5
2020 B=17: A=18-G(J)
2030 FOR I=A TO B
2040 LOCATE (J-1)*6+2,I: PRINT CHR$(143)
2050 LOCATE (J-1)*6+3,I: PRINT CHR$(143)
2060 NEXT I
2070 NEXT J
2080 RETURN
```

Lista de variables:

A = Línea de inicio del gráfico

B = Línea final del gráfico

F = Porcentajes

FM = Valor máximo

G = Porcentaje transformado

I = Indice variable

J = Indice variable

Descripción del programa:

Línea 10 - 90	Título y aclaraciones
Línea 95	Dimensionado
Línea 100 - 140	Introducción informaciones INPUT
Línea 150 - 160	Borrado de pantalla y dibujo de una línea horizontal
Línea 180	Búequeda del porcentaje mayor
Línea 200 - 210	Conversión de los porcentajes de manera que el valor máximo aproveche el espacio disponible en la pantalla
Línea 230	Bifurcación a subrutina en línea 200, para dibujar el gráfico
Línea 250 - 270	Impresión de los textos PSOE, AP, CIU, PCE y OTROS así como de los porcentajes debajo del dibujo
Línea 280	Final del programa principal

Línea 2000 - 2090 Subrutina para el trazado de las barras

Línea 2010-2020:
Determinación de las direcciones inicial y final de la barra

Línea 2030-2070:
Trazado de blancos inversos en los puntos fijados mediante LOCATE

Línea 2090 : Retorno

Línea 2030-2060:
Trazado de blancos inversos en los puntos fijados mediante POSITION

Línea 2080:
Vuelta

Epílogo

En los capítulos precedentes hemos intentado mostrar cómo pueden resolverse problemas típicos de la escuela con programas BASIC relativamente sencillos.

Desearía insistir en que los ejemplos escogidos tienen únicamente carácter de muestra, y que no se le ha dado concedido especial importancia a los programas óptimos.

El lector debe comprender rápidamente el funcionamiento de los programas. De este modo será capaz de encontrar la vía de resolución de otros problemas, que frecuentemente se asemejarán a los que aquí presentamos.

Lo fundamental a la hora de utilizar un ordenador no es el ordenador en sí, sino la capacidad que demuestra el usuario para tratar determinados problemas, de forma que pueda ofrecerle al ordenador el tipo de resolución más adecuada para un programa.

No es necesario que un programa funcione perfectamente a la primera (aun cuando esto, naturalmente, sea muy positivo); el usuario debe aprender cómo estructurar mentalmente un problema dado, para que el ordenador pueda resolverlo :

La causa es que el ordenador no sirve para nada si nosotros, los usuarios, no conocemos la vía de resolución con anterioridad.

Si alguna vez ocurriese que un programa no funciona tal y como nosotros lo habíamos previsto, informándonos mediante un mensaje de error y la interrupción del programa, siempre nos ayudará el echar una ojeada al manual del ordenador.

A aquellos lectores, a los que los ejemplos presentados les han parecido demasiado sencillos, les indicaremos que este libro ha sido concebido en primera instancia para principiantes del BASIC y que no hemos querido superar sus posibilidades a través de una complejidad excesiva de la obra. Para consuelo de los primeros, queremos anunciar la pronta publicación de un segundo tomo de esta obra, donde se estudiarán problemas más complejos, especialmente de tipo matemático.

A

B

C

D

E

F

G

H

I

L

M

N

O

P

P

Q

R

R

S

T

U

V



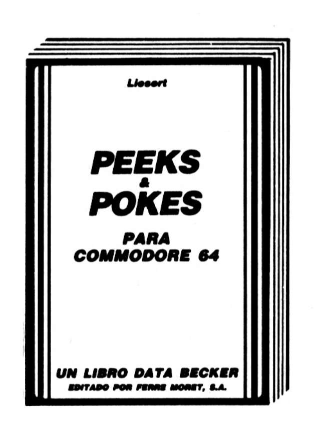
ROBOTICA PARA SU COMMODORE 64, 230 pág.
P.V.P. 2.800,- ptas.

En el libro de los robots se muestran las asombrosas posibilidades que ofrece el CBM 64, para el control y la programación, presentadas con numerosas ilustraciones e intuitivos ejemplos. El punto principal: Cómo puede construirse uno mismo un robot sin grandes gastos. Además, un resumen del desarrollo histórico del robot y una amplia introducción a los fundamentos cibernéticos.
Gobierno del motor, el modelo de simulación, interruptor de pantalla, el Port-Usuario cómodo del modelo de simulación, Sensor de infrarrojos, concepto básico de un robot, realimentación unidad cibernética, Brazo prensor, Oír y ver.



MANUAL ESCOLAR PARA SU COMMODORE 64, 351 pág.
P.V.P. 2.800,- ptas.

Este libro, escrito especialmente para escolares de grado medio y superior, contiene muchos interesantes programas de aprendizaje para solucionar problemas, descritos detalladamente y de manera fácilmente comprensible. Facilitan un aprendizaje intensivo y ameno, con, entre otros, los siguientes temas: Teorema de pitágoras, progresiones geométricas, palanca mecánica, crecimiento exponencial, verbos irregulares, ecuaciones de segundo grado, movimientos de péndulo, formación de moléculas, aprendizaje de vocablos, cálculo de interés y su capitalización. Una corta repetición de los elementos BASIC más importantes y una introducción a los rasgos esenciales del análisis de problemas, entre otros, completan el conjunto.



PEEKS y POKES, 177 pág.
P.V.P. 1.600,- ptas.

Con importantes comandos PEEK y POKE se pueden hacer también desde el Basic muchas cosas, para las que se necesitarían normalmente complejas rutinas en lenguaje máquina. Este libro explica de manera sencilla el manejo de PEEKs y POKEs. Con una enorme cantidad de POKEs importantes y su posible aplicación. Para ello se explica perfectamente la estructura del Commodore 64: Sistema operativo, interpretador, página cero, apuntadores y stacks, generador de caracteres, registros de sprites, programación de interfaces, desactivación del interrupt. Además una introducción al lenguaje máquina. Muchos programas ejemplo.



TODO SOBRE EL FLOPPY 1541, 482 pág.
P.V.P. 3.200,- ptas.

La obra Standard del floppy 1541, todo sobre la programación en disquettes desde los principiantes a los profesionales, además de las informaciones fundamentales para el DOS, los comandos de sistema y mensajes de error, hay varios capítulos para la administración práctica de ficheros con el FLOPPY, amplio y documentado Listado del Dos. Además un filón de los más diversos programas y rutinas auxiliares, que hacen del libro una lectura obligada para los usuarios del Floppy.



MANTENIMIENTO Y REPARACION DEL FLOPPY 1541,
200 pág.
P.V.P. 2.800,- ptas.

Saberse apañar uno mismo, ahorra tiempo, molestias y dinero, precisamente problemas como el ajuste del floppy o reparaciones de la platina se pueden arreglar a menudo con medios sencillos. Instrucciones para eliminar la mayoría de perturbaciones, listas de piezas de recambio y una introducción a la mecánica y a la electrónica de la unidad de disco, hay también indicaciones exactas sobre herramientas y material de trabajo. Este libro hay que considerarlo en todos sus aspectos como efectivo y barato.



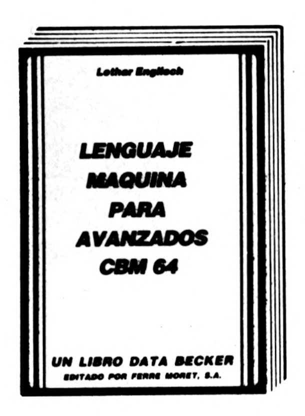
EL MANUAL DEL CASSETTE, 190 pág.
P.V.P. 1.600,- ptas.

Un excelente libro, que le mostrará todas las posibilidades que le ofrece su grabadora de casettes. Describe detalladamente, y de forma comprensible, todo sobre el Datassette y la grabación en cassette. Con verdaderos programas fuera de serie: Autostart, Catálogo (¡busca y carga automáticamente!), backup de y a disco, SAVE de áreas de memoria, y lo más sorprendente: un nuevo sistema operativo de cassette con el 10-20 veces más rápido FastTape. Además otras indicaciones y programas de utilidad (ajuste de cabezales, altavoz de control).



LENGUAJE MAQUINA PARA
COMMODORE 64, 1984, 201 pág.
P.V.P. 2.200,- ptas.

¡Por fin una introducción al código máquina fácilmente comprensible! Estructura y funcionamiento del procesador 6510, introducción y ejecución de programas en lenguaje máquina, manejo del ensamblador, y un atractivo muy especial: ¡un simulador de paso a paso escrito en BASIC!



LENGUAJE MAQUINA PARA AVANZADOS
CBM 64, 1984, 206 pág.
P.V.P. 2.200 ptas.

¿Ud. ha logrado iniciarse en código máquina? Entonces el «nuevo English» le enseñará cómo convertirse en un profesional. Naturalmente con muchos programas ejemplo, rutinas completas en código máquina e importantes consejos y trucos para la programación en lenguaje máquina y para el trabajo con el sistema operativo.



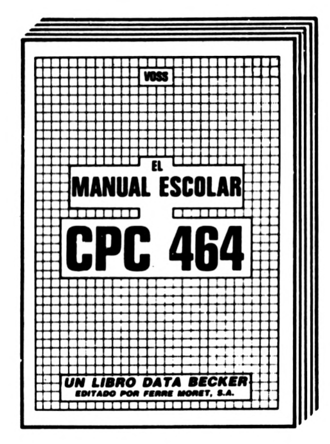
EL ENSAMBLADOR

Este libro ofrece al programador interesado una introducción fácilmente comprensible para los tan extendidos Assembler PROFI-ASS, SM MAE y T.EX.ASS. con consejos y trucos de gran utilidad, indicaciones y programas adicionales. Al mismo tiempo sirve de manual orientado a la práctica, con aclaraciones de conceptos importantes e instrucciones.
250 páginas. 2.200,- ptas.



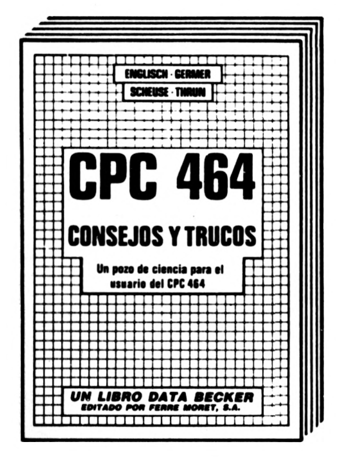
64 EN EL CAMPO DE LA TECNICA Y LA CIENCIA, 296 pág.
P.V.P. 2.800,- ptas.

Ofrece un campo fascinante y amplio de problemáticas científicas. Para esto el libro contiene muchos listados interesantes: Análisis de Fornier y síntesis, análisis de redes, exactitud de cálculo, formateado de números, cálculo del valor PH, sistemas de ecuaciones diferenciales, modelo ladrón presa, cálculo de probabilidad, medición de tiempo, integración, etc.



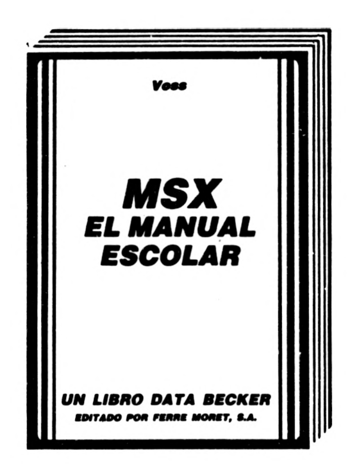
CPC-464 EL LIBRO DEL COLEGIO
P.V.P. 2.200,- ptas.

Escrito para alumnos de los últimos cursos de EGB y de BUP, este libro contiene muchos programas para resolver problemas y de aprendizaje, descritos de una forma muy compleja y fácil de comprender. Teorema de Pitágoras, progresiones geométricas, escritura cifrada, crecimiento exponencial, verbos irregulares, igualdades cuadráticas, movimiento pendular, estructura de moléculas, cálculo de interés y muchas cosas más.



CPC-464 CONSEJOS Y TRUCOS
P.V.P. 2.200,- ptas.

Ofrece una colección muy interesante de sugerencias, ideas y soluciones para la programación y utilización de su CPC-464: Desde la estructura del hardware, sistema de funcionamiento - Tokens Basic, dibujos con el joystick, aplicaciones de ventanas en pantalla y otros muchos interesantes programas como el procesamiento de datos, editor de sonidos, generador de caracteres, monitor de código máquina hasta listados de interesantes juegos.



MSX EL MANUAL ESCOLAR
P.V.P. 2.800,- ptas.

Escrito para alumnos de los últimos cursos de EGB y de BUP, este libro contiene muchos programas para resolver problemas y de aprendizaje, descritos de una forma muy completa y fácil de comprender. Teorema de Pitágoras, progresiones geométricas, escritura cifrada, crecimiento exponencial, verbos irregulares, igualdades cuadráticas, movimiento pendular, estructura de moléculas, cálculo de interés y muchas cosas más.



MSX GRAFICOS Y SONIDOS, 250 pág.
P.V.P. 2.800,- ptas.

Las computadoras MSX no sólo ofrecen una relación precio/rendimiento sobresaliente, sino que también poseen unas cualidades gráficas y de sonido excepcionales. Este libro expone las posibilidades de los MSX de forma completa y fácil. El texto se completa con numerosos y útiles programas ejemplo.



MSX PROGRAMAS Y UTILIDADES, 1985, 194 pág.
P.V.P. 2.200,- ptas.

El libro contiene una amplia colección de importantes programas que abarcan, desde un desensamblador hasta un programa de clasificaciones deportivas. Juegos superemocionantes y aplicaciones completas. Los programas muestran además importantes consejos y trucos para la programación. Estos programas funcionan en todos los ordenadores MSX, así como en el SPECTROVIDEO 318 328. ESTRACTO DEL CONTENIDO: Volcado memoria hexadecimal. Editor gráficos. Editor de sonido. Escritura de ordenador. Lista referencia de variables. Calendario. Desensamblador. ADMINISTRACION de una colección de discos L.P. HOLLOW - JUEGO DE LAS CEREZAS. DIAGRAMAS DE BARRAS. TABLAS DEPORTIVAS.



ZX SPECTRUM EL MANUAL ESCOLAR
P.V.P. 2.200,- ptas.

Escrito para alumnos de los últimos cursos de EGB y de BUP, este libro contiene muchos programas para resolver problemas y de aprendizaje, descritos de una forma muy completa y fácil de comprender. Teorema de Pitágoras, progresiones geométricas, escritura cifrada, crecimiento exponencial, verbos irregulares, igualdades cudráticas, movimiento pendular, estructura de moléculas, cálculo de interés y muchas cosas más.



ZX SPECTRUM CONSEJOS Y TRUCOS, 211 pág.
P.V.P. 2.200,- ptas.

Una interesante colección se sugestivas ideas y soluciones para la programación y utilización de su ZX ESPECTRUM. Aparte de muchos peeks, pokes y USRs hay también capítulos completos para, entre otros, entrada de datos asegurado sin bloqueo de ordenador, posibilidades de conexión y utilización de microdrives y lápices ópticos programas para la representación de diagramas de barra y de tarta, el modo de utilizar óptimamente ROM y RAM.



METODOLOGIA DE LA PROGRAMACION
P.V.P. 2.200,- ptas.

El primer libro recomendado para escuelas de enseñanza de informática y para aquellas personas que quieren aprender la programación. Cubre las especificaciones del Ministerio de Educación y Ciencia para Estudios de Informática. Realizado por un alto mando del ejército Español, un Dr. Ingeniero y Diplomado en Informática y profesor de la UNED y por un oficial técnico especialista en informática de gestión. Utilizado en todos los institutos politécnicos del ejército español. Es un seguro candidato a ediciones en lengua inglesa, alemana y francesa. Es el primer libro que introduce a la lógica del ordenador. Es un elemento de base que sirve como introducción para la programación en cualquier otro lenguaje. No se requieren conocimientos de programación ni siquiera de informática. Abarca desde los métodos de programación clásicos a los más modernos.



TODO SOBRE EL NUEVO COMMODORE 128

P.V.P. 2.200,- ptas.

El Libro de Primicias del Commodore 128 no ofrece solamente un resumen completo de todas las características y rendimientos del sucesor del C-64 y con ello una importante ayuda para su adquisición. Muestra, además, todas las posibilidades del nuevo equipo en función de sus tres modos de operación.

Entre otros se describen el hardware, los modos de operación: modo 64, modo 128 y modo CP/M, las configuraciones de memoria, la disposición de la página cero, trabajos con dos pantallas, modo de 80 caracteres, Basic V 7.0: comandos de gráficos y sonidos, comandos de control, periféricos rápidos (1571) etc.

RESPUESTA
COMERCIAL

F.D. Autorización 6975
(B.O. de Correos N.º 80 de 26-7-85)

HOJA PEDIDO
DE LIBRERIA

NO NECESITA
SELLOS

A franquear
en destino

FERRE MORET, S.A.

Apartado N.º 551. F.D.
08080 BARCELONA

RESPUESTA
COMERCIAL

F.D. Autorización 6975
(B.O. de Correos N.º 80 de 26-7-85)

HOJA PEDIDO
DE LIBRERIA

NO NECESITA
SELLOS

A franquear
en destino

FERRE MORET, S.A.

Apartado N.º 551. F.D.
08080 BARCELONA

Puesta al día de datos

EDITORIAL FERRER MORET, S.A. mantiene vivo y amplía el contenido informativo de sus libros y programas, mediante el envío de un servicio de puesta al día, junto con una síntesis noticiosa de la actualidad y perspectivas de la realidad informática española.
Agradecemos cualquier sugerencia o crítica que desee formular y que nos ayude a mejorar las ediciones. Muchas gracias.

¿Qué añadiría?

....

....

¿Qué suprimiría?

....

....

Observaciones

....

....

Título del libro

....

Nombre

....

Dirección Tfno.

....

Código Postal y Población Provincia

....

UN SERVICIO GRATUITO

Información

FERRE MORET, S.A. cuenta con un amplio fondo de libros y Software y mantiene un servicio de información por correo sobre las novedades que edita.
Agradecemos nos indique los temas que representan para Vd. mayor interes.

Libros

- ☐ ATARI
- ☐ COMMODORE
- ☐ SINCLAIR
- ☐ MSX
- ☐ LENGUAJES
- ☐ IBM
- ☐ AMSTRAD
- ☐ APPLE
- ☐ SOFTWARE

Si está interesado en recibir alguno de estos servicios, rellene y envíe la tarjeta correspondiente; no necesita franqueo. Muchas gracias.

DESEO RECIBIR EL LIBRO ....

EL PROGRAMA ....

☐ Adjunto cheque ☐ Contra reembolso

....

Nombre

....

Dirección Tfno.

....

Código Postal y Población Provincia

....

UN SERVICIO GRATUITO

EL CONTENIDO:

Escrito para alumnos de los últimos cursos de EGB y de BUP, este libro contiene multitud de programas para resolver problemas y de aprendizaje, descritos de una forma muy completa y fácil de comprender. Proporcionan un aprendizaje intensivo, a la vez que divertido, con el CPC 464/6128.

Del contenido:

- Teorema de Pitágoras
- Progresiones geométricas
- Escritura cifrada
- Crecimiento exponencial
- Verbos irregulares
- Igualdades cuadráticas
- Movimiento pendular
- Estructura de moléculas
- Repaso de vocablos
- Cálculo de intereses

Una pequeña panorámica sobre los fundamentos del tratamiento informático, un somero repaso a los principales elementos del BASIC y una introducción en los aspectos básicos del análisis de problemas completan el libro.

ESTE LIBRO HA SIDO ESCRITO POR:

Werner Voss, profesor de estadística en la Universidad de Bochum. Multitud de publicaciones dentro del campo de la estadística y del tratamiento informático.

ISBN 84-86437-15-6

Voss / El Manual escolar CPC 464/6128



AMSTRAD
CPC
OCR
MÉMOIRE ÉCRITE
MEMORY ENGRAVED
MEMORIA ESCRITA
AC
ME
https://acpc.me/